DNA PROPERTIES OF COMPUTATIONALLY MAPPED NUCLEOSOME POSITIONS

Master Thesis

Systems biology master program

Vilnius university

STUDENT NAME: Indiras Maziukas

STUDENT NUMBER: 2011286

SUPERVISOR: doc. dr. Erinija Pranckevičienė

SUPERVISOR DECISION: ……

FINAL GRADE ……

DATE OF SUBMISSION: 15 May 2023

CONTENTS

LIST OF ABBREVIATIONS

HMM – Hidden Markov Model

MHMM – Multivariate Hidden Markov Model

HHMM – Hierarchical Hidden Markov Model

dHMM – Duration Hidden Markov Model

PC – Principal Component

ROC – receiver operator characteristic

ATAC-Seq – Assay for Transposase-Accessible Chromatin with high-throughput sequencing

ChIP-Seq – Chromatin immunoprecipitation followed by sequencing

MNase-Seq – Micrococcal nuclease sequencing

DNase-Seq – DNase I hypersensitive sites sequencing

HGT – horizontal gene transfer

HGHMM – Hierarchical generalized hidden Markov models

NFR – nucleosome-free region

INTRODUCTION

Nucleosomes play a part in gene regulation by controlling access to specific parts of DNA via granting access for transcription or restricting it. Understanding the factors that influence nucleosome positioning and how to identify their positions can provide insights into how genes are regulated.

Nucleosomes are structural units that package DNA in eukaryotes. Each nucleosome consists of a histone octamer with 147 bp DNA wrapped around it. Nucleosomes and their positional patterns are at least partially responsible for chromatin structure and gene expression.

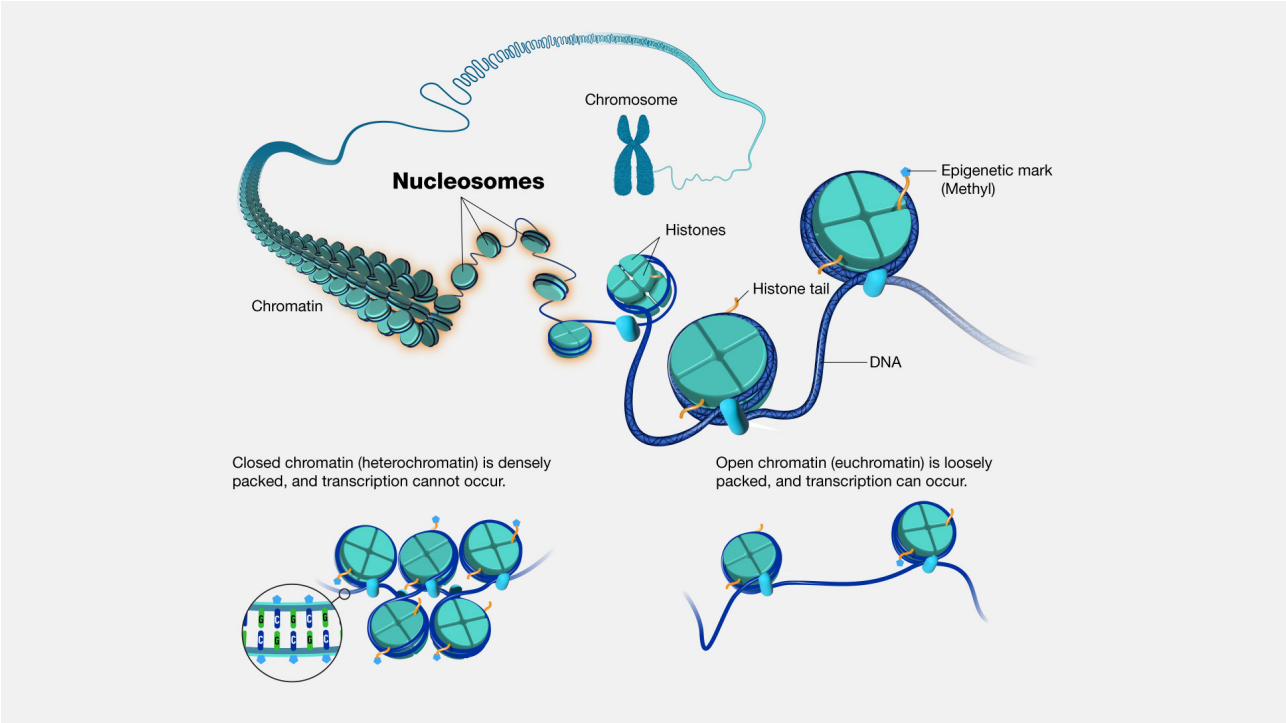


Figure 1. Nucleosomes within DNA. Source: https://www.genome.gov/genetics-glossary/Nucleosome (accessed 14 May 2023)

Nucleosomes and DNA together make up chromatin. Chromatin itself is dynamic - remodelling is happening constantly across the whole genome (Sananbenesi and Fischer, 2015). It is a process that changes the architecture of chromatin and therefore allows and controls access to genomic DNA and its transcription. (Zhou et al., 2016) Chromatin remodelling happens in response to environmental stress and other conditions by sliding, spacing out, ejecting, and adding nucleosomes.

Certain patterns of dinucleotides in nucleosomal DNA show very high stability and affinity to the histone octamer. Such patterns can be termed as packing (Pranckevičienė et al., 2020). In a stack

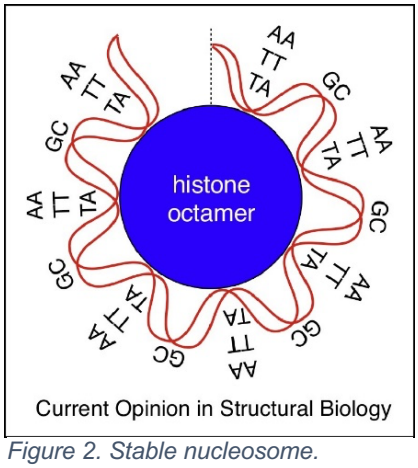


Figure 2. Stable nucleosome. (Onufriev and Schiessel, 2019)

of aligned nucleosomal sequences these patterns manifest statistically as peaks (Supplementary figure 3) of dinucleotide occurrence frequency along a nucleosomal sequence.

Some tools exist that can be used to identify nucleosome positions solely based on the DNA sequences (Pranckevičienė et al., 2020, Pranckevičienė et al., 2022). Generally, these methods are not very accurate. Several tools have been developed to predict nucleosome positions using Hidden Markov Models (HMMs) that are claimed to be more accurate. In this thesis we focus on Hidden Markov Model algorithms and tools for nucleosome mappings.

HMMs are probabilistic models that can capture the underlying structure of a sequence (Eddy, 2004). The model can be applied to infer probability of a nucleosome being present at each position of the DNA sequence by analysing this sequence. Several studies, such as *NuPoP* (Xi et al., 2010), *Hierarchical Generalized Hidden Markov Models* (Moser and Gupta, 2012), have shown that HMMs can accurately predict nucleosome positions in a variety of organisms. By comparing HMM predictions in different DNA sequences, such as healthy and diseased states, cell types and responses to stimuli, results can help identify differences in nucleosome positioning that are associated with gene expression and chromatin structure.

An advantage that HMMs have over other nucleosome prediction methods (such as support vector machines or genomic signal processing (Peckham et al., 2007)) is their ability to model the dependence of nucleosome positioning on the sequence context. This means that the model can consider the specific DNA sequence in addition to other factors such as histone modifications. As more genomic data becomes available (Zhao et al., 2019), HMMs are likely to become even more powerful in predicting nucleosome positions because the amount of training sequences will increase. HMMs are often used to infer nucleosome positions within DNA, therefore it can prove useful to have multiple tools at our disposal to access easily (Tsui, 2013). At the moment we can find several useful tools for such analysis using HMMs. These tools include *HMMRATAC* (Tarbell and Liu, 2019), *NuPoP* (Xi et al., 2010), NucHMM (Fang et al., 2021). While they are available, they have to be discovered and set up separately, with no way to have a complete workflow easily accessible. While this is possible to do, it can cause inconveniences and slowdowns when working with the tools for the first time or when tools do not have an easy-to-use interface which can slow down the research process and create unnecessary confusion when navigating between multiple platforms and tools. For such cases, we have decided to find useful tools to work with HMMs for nucleosome position prediction and integrate them together via an extensible custom software tool with already existing tools to create a complete workflow. This can provide a selection of tools easily with the possibility to compare results or to find the multiple approaches with data available.

AIM AND TASKS

The aim of this thesis is to develop a Hidden Markov Model based computational nucleosome mapping tool.

This aim comprises of the following tasks:

- Create Hidden Markov model and implement an HMM based algorithm programmatically to predict nucleosome positions in a DNA sequence and provide for the model for use
- Use known human and mouse nucleosomal sequences to train the HMM and use it to infer DNA properties
- Test the trained model to map nucleosomes and calculate accuracy using 204 nucleosomal sequences in which a position of nucleosome is known
- Integrate the developed HMM based tool into existing Galaxy instance

LITERATURE REVIEW

Nucleosome positioning sequences play an important role in nucleosome positioning along other factors, such as epigenetic DNA modifications, plasticity, and chromatin remodelling complexes. The patterns of DNA sequences forming nucleosomes have been statistically categorized roughly into two groups – packing and regulatory – that differ in their dinucleotide composition and spatial dinucleotide frequency. The plasticity of DNA in these sites is an important factor that influences the addition of new, advantageous genes and the deletion of unnecessary ones. In the context of nucleosomes, packaging and regulating the expression of genes can indirectly influence the selection of the genes. The processes are closely related to HGT barrier and the size of the genome (Sela et al., 2018). However, positional accuracy problems still persist when mapping nucleosomes, which can impede research and lead to relatively inaccurate data - the results can have a success rate of 50% (Zhou et al., 2016). Further analysis of the connection between regulatory chromatin sites and DNA plasticity and rigidity is needed to understand their impact on the genome. For this thesis, a particular interest is placed on the effectiveness of Hidden Markov Models for such use. Various tools implement HMMs in different ways, although there is no integrated complete solution. Tools can be found on several platforms, such as NucPosDB (Shtumpf et al., 2022), Nucleosome Dynamics (Buitrago et al. 2019). Alongside the tools available, there is research that shows the results that such models can provide, and apart from the models themselves, there is a possibility to connect such tools in a unified workflow.

Hidden Markov Models

A Hidden Markov Model is a statistical model that can be used to analyse sequences of observations. A sequence is produced through a stochastic progression of multiple states. Each state represents a specific collection of labels that provide detailed information about a small portion of the subject or its constituent part, while observations are the visible data points produced by the states. HMMs are particularly useful when analysing sequences in which the states or categories of the underlying process generating the sequence are not directly observable. The goal is to define the next state by the current one (Eddy, 2004).

In an HMM, the states of the underlying process are hidden or unknown, but the observations are visible or observable. The model consists of a set of states, a set of observations, and a set of probabilities that describe the relationships between the states and the observations.

The key to using HMMs is to learn the model parameters from data, which involves estimating the probabilities of transitioning between states and emitting observations from each state. This is typically done using an algorithm called the Baum-Welch algorithm, which is a type of Expectation-

Maximization algorithm that estimates the maximum likelihood parameters of the model (Eddy, 2004).

HMMs are used in a wide range of applications, including speech recognition, bioinformatics, and natural language processing. In bioinformatics, HMMs are particularly useful for analysing biological sequence data, such as DNA or protein sequences, and for identifying functional regions within these sequences. It can commonly be seen that one way to analyse the results of an HMM is the Viterbi algorithm which is used to find the most likely sequence of hidden states that generated a sequence of observations. In our case HMMs are particularly useful because they can capture the periodicity and spatial dependencies that are characteristic of nucleosome positioning. (Durbin et al., 1998)

The model assumes that the sequence of observations was generated by an underlying process with a set of states, but the states themselves are hidden and not directly observable. Each state emits an observation, and the probability of emitting a particular observation depends on the current state of the process.

Variations of HMMs

Profile HMMs are a type of HMM that is used to represent a multiple sequence alignment, which is a collection of related sequences that have been aligned to identify conserved regions. Profile HMMs use the position-specific scoring matrix of the multiple sequence alignment to represent the emission probabilities of the states. This allows the model to capture the pattern of conservation and variation across the aligned sequences. (Durbin et al., 1998)

Pair HMMs are used to align two sequences, such as a DNA or protein sequence and a reference sequence. Pair HMMs use two different sets of states, one for the query sequence and one for the reference sequence, and incorporate information about the substitution, insertion, and deletion probabilities of aligning two sequences. (Durbin et al., 1998)

Multivariate hidden Markov models (MHMMs) extend the basic HMM framework to handle multiple sequences of observations. In an MHMM, each observation is a vector of measurements rather than a single scalar value. MHMMs allow the joint probability distribution of the observations to be modelled as a multivariate Gaussian distribution, and the transition probabilities between hidden states are modelled using a regular HMM. (Visser and Speekenbrink, 2022)

Hierarchical hidden Markov models (HHMMs) extend the HMM framework to handle sequences of HMMs. In an HHMM, each hidden state corresponds to a separate HMM, and the transitions between the hidden states are themselves modelled as an HMM. HHMMs allow for a hierarchical structure of

hidden states, where each level of the hierarchy corresponds to a different level of abstraction or granularity in the underlying process that generates the observations (Fine et al., 1998).

The Galaxy Platform

The Galaxy platform is an open-source bioinformatics platform designed to make it easier for researchers to perform data analysis and visualization of large biological datasets, such as DNA sequencing data, proteomics data, and other omics data. Galaxy was first developed in 2005 and has since been widely adopted by the scientific community. (Galaxy Community, 2022)

The Galaxy platform provides a web-based interface that allows researchers to access a wide range of bioinformatics tools and workflows, without requiring them to have specialized computing or programming expertise. Users can upload their own data, perform analysis using a wide range of tools, and visualize and share their results. The platform also enables reproducibility by allowing researchers to share their workflows and data with others.

The Galaxy platform has been used in a wide range of scientific research, including genomics, transcriptomics, metagenomics, and proteomics. It has been used to analyse data from a wide range of organisms, including humans, animals, plants, and microbes. The platform is constantly evolving, with new tools and features being added regularly by the community of developers and users.

Investigated studies

Hierarchical generalized hidden Markov models (HGHMM)

The study (Moser and Gupta, 2012) discusses the use of hidden Markov models for statistical inference from genome tiling arrays, developing a hierarchical model robust to various sources of probe variability and measurement error and an explicit state duration model. The sources of variability are mainly from length restrictions and unknown length duration of states, exponential forgetting (Moser and Gupta, 2012). In the context of tiling-array data, which are series of hybridised short overlapping probes that cover the whole genome (Gupta, 2007), HMMs are useful for detecting true protein-DNA interactions because the spatially dependent structure of the tiling array suggests that models explicitly incorporating this dependence are more powerful (Moser and Gupta, 2012).

However, HMMs are not directly suitable for assessing length-constrained features such as nucleosomes because they induce exponentially decaying state length distributions.

$$(1)\ \lim_{n\to+\infty} || \rho_n - \rho'_n || = 0\ almost\ surely$$

Equation 1. Exponential forgetting of prediction filter or loss of memory of HMM. Start any two different initial state probability vector p_0 and p'_0 and after applying the same sequence of matrices, they generate two sequence of filtered state probability ρ_n and ρ'_n. The distance of two sequences goes to 0 asymptotically almost surely (Ye et al., 2017).

To address this problem, a generalized Bayesian framework was introduced for statistical inference from genome tiling arrays, developing a hierarchical model that is robust to previously mentioned sources of probe variability and measurement error and an explicit state duration (Moser and Gupta, 2012).

Wavelets are mathematical functions that are used to represent complex signals by decomposing them into smaller blocks. These algorithms use wavelet analysis, which can help remove noise from data, to determine signals, which are used to model the probability that DNA sequence is part of a nucleosome via logistic regression, which is a method that analyses the relationship between categorical and independent variables via using a logistic function to model the probability of an event occurrence. The predicted logits, which are the natural logarithm of the odds of an event occurring and in this case are the outputs, also known as N-scores, are used to classify each sequence as a linker or nucleosome. These methods have been compared to other nucleosome classification approaches using an ROC-score, the area under a receiver operator characteristic (ROC) curve. The article concludes that combining sequence information with nucleosome positioning data can improve the accuracy of discovering transcription factor binding sites in complex organisms.

The study (Moser and Gupta, 2012) describes two methods for analysing nucleosome positioning experiments using probe-specific models. The first method uses a hierarchical hidden Markov model approach to model the spatial dependence between probes, allowing for flexible modelling of the distribution of latent states. The second method uses a two-stage approach to determine sequence-based characteristics that predict nucleosome positioning. A two-state hierarchical HMM is used, where at the coarsest level, different segment types may potentially have different nucleotide compositions. The increase in predictive power is tested via using data where nucleosome positions are known. The study's results indicated that the HGHMM-based methods had had a smaller discrepancy from real percentage of nucleosomal regions when compared to methods by (Yuan and Liu, 2008) and (Segal et al., 2006). HGHHM proved to be better in predicting nucleosome-rich regions than nucleosome-free regions, but the article suggests that while sequence factors are generally indicative of differences between nucleosomal and nucleosome-free regions, other chromatin measurements may need to be integrated for maximal predictive efficiency. They also showed that A/T-containing dimers and trimers are the top contributors to nucleosome positioning, and the 3rd principal component (PC), which is most strongly correlated with nucleosome positioning, depends heavily on C- and G-containing k-mers, suggesting that there may be mechanisms at work other than the rigidity of the DNA alone in positioning nucleosomes.

Genome-Scale Identification of Nucleosome Positions in S. cerevisiae

In this approach, a HMM is trained on a set of nucleosome and linker sequences, which is used to generate a model that can predict the probability of a given sequence being a nucleosome or linker. The model is then used to predict the probability of each nucleotide being in a nucleosome or linker state along the DNA sequence. This probability profile can be thresholded to identify the start and end positions of nucleosomes and linkers. The main advantage of using HMMs is that they can capture long-range dependencies between nucleotide positions, which is important for accurately predicting nucleosome and linker boundaries. (Yuan et al., 2005)

Genomic Sequence Is Highly Predictive of Local Nucleosome Depletion

In this study (Yuan and Liu, 2008), a computational approach is presented to extract sequence features associated with nucleosome binding. The approach involves the use of the wavelet transformation to extract periodicity features and then uses a statistical model to select features associated with nucleosome positioning. The model has a significantly improved performance relative to previous studies in predicting genome-wide nucleosome positions. The study also identified long-range (>100 bp) or sequence-independent signals that are important for nucleosome positioning. The approach was used to analyse human genomic sequences, and the results were in good agreement with experimental data. (Yuan and Liu, 2008)

Although computational methods that extract sequence features associated with nucleosome binding have been developed, the prediction accuracy is only modestly higher than random guessing, suggesting that there may be additional long-range (>100 bp) or sequence independent signals that are important for nucleosome positioning undiscovered by current models. Their method first makes use of the wavelet transformation (Addison, 2005) to extract periodicity features and then uses a statistical model to select features associated with nucleosome positioning. The model was able to predict *in vivo* nucleosome-enriched or nucleosome-depleted regions, negative correlation between promoter nucleosome occupancy and global transcription rates, depletion of nucleosomes at regulatory elements, and mutation of short DNA sequences that only leads to gradual changes of nucleosome occupancy.

The article (Yuan and Liu, 2008) also discusses the development of a Hidden Markov Model to predict genome-wide nucleosome positions from tiling array data. The model uses the N-score to quantify the sequence preference of nucleosome binding and takes into account that the positions of neighbouring nucleosomes interfere with each other. The model predicts non-overlapping local peaks of the N-scores, representing the predicted nucleosome positions. The performance of the model was evaluated against non-chromosome III nucleosome positions, and it was found to have a lower false negative rate and false positive rate than random guessing. The article also discusses the enrichment of regulatory elements in low N-score regions and the reduced nucleosome occupancy at the TATA box.

HMMER

HMMER is a tool for searching sequence databases for homologous protein or nucleotide sequences, using profile hidden Markov models. Although *HMMER* is primarily used for protein sequence analysis, it can also be used for nucleotide sequence analysis. (Potter et al., 2018; Mistry et al., 2013)

One way to use *HMMER* for nucleosome position detection is to create an HMM based on the nucleosome sequence motif, which can be used to search genomic DNA for nucleosome positions. The nucleosome sequence motif is a pattern of DNA sequence that is characteristic of regions that are wrapped around a nucleosome.

To create an HMM for nucleosome position detection, one approach is to train the model using a set of known nucleosome positions. This can be done by first extracting DNA sequences around the known nucleosome positions, and then using these sequences to generate a multiple sequence alignment. The multiple sequence alignment can then be used to build a profile HMM using *HMMER*. Once the HMM is created, it can be used to search genomic DNA for potential nucleosome positions. The HMMER search will return a list of genomic regions that match the HMM profile. These regions can then be further analysed to determine whether they are likely to be nucleosome positions.

It's worth noting that HMMER is just one tool that can be used for nucleosome position detection, and there are many other methods and software packages available, such as *HMMRATAC* (Tarbell and Liu, 2019), *NuPoP* (Xi et al., 2010). Additionally, nucleosome position detection can be a challenging problem, and multiple methods are often used in combination to improve accuracy and reliability (Wheeler and Eddy, 2013).

Overview of investigated studies

Overall, studies show that Hidden Markov Models are not only a subject that is being researched to use in future studies. HMMs have already proven useful in multiple studies, including nucleosome position prediction. Such use of HMMs not only provide an improvement in accuracy or speeds, but it also allows to take note of other features that are not easily detectable. Such models have some intrinsic properties that can be applied to predict nucleosome positions in a way that matches what the provided sequences do not show initially.

Investigated tools

NucHMM

The study (Fang et al., 2021) introduces a novel computational method called *NucHMM*, which integrates a hidden Markov model with the characteristics of nucleosome organization to identify cell

type-specific functional nucleosome states. The method is tested on publicly available data and offers insights into the interplay between nucleosome organization and the splicing process. (Fang et al., 2021)

The algorithm is composed of three modules: initialization, training, and functioning. The training module trains multiple Hidden Markov Models and selects the best model based on the smallest Bayesian Information Criterion score (Fang et al., 2021). The functioning module defines functional nucleosome states based on genomic location, average number of nucleosomes, nucleosome phasing and spacing, and nucleosome positioning. The model is trained using ChIP-seq and MNase-seq data from (The ENCODE Project Consortium, 2012) and is able to identify functional nucleosome states. The algorithm is able to identify 11 functional nucleosome states, each with a specific set of histone modifications, and can be used to determine the splicing potential of specific nucleosome states. The research implies that nucleosomes bearing H3K36me3, and H3K79me2 histone tail modifications may have a significant impact on the process of skipping exon processing. The tool was tested on publicly available data of MCF7, H1, and IMR90 cells (The ENCODE Project Consortium, 2012).

NuPoP

The article (Xi et al., 2010) presents a method for modelling chromosomal DNA sequences using duration Hidden Markov Models (dHMMs). The dHMM used models two states, nucleosome, and linker, where the nucleosome state has a fixed length of 147 base pairs and the linker state has a variable length. The models were trained using a set of 503,264 yeast nucleosome DNA reads from 454 pyrosequencing, and the method was used to estimate the nucleosome occupancy and histone binding affinity score at a specific position in the DNA sequence. Two models were trained - one for nucleosomes (trained with non-redundant nucleosome sequences – 4th order time-dependent Markov chain) and one for linker regions (8090 reads-free regions – homogeneous 4th order Markov chain). The optimal path was found using the Viterbi algorithm, and the nucleosome occupancy score was estimated using forward and backward algorithms. The authors stated that the dHMM model provided a more accurate result for nucleosome positions than previous methods. (Xi et al., 2010)

nhmmer

One more specific tool from the *HMMER* suite that can be used for nucleosome position prediction is *nhmmer*.

As other tools fail to produce good results, profile hidden Markov models are introduced as a more sensitive and probabilistic approach to sequence comparison. They calculate the signal of homology

based on the more powerful Forward/Backward HMM algorithm that computes not just one best-scoring alignment, but a sum of support over all possible alignments. (Wheeler and Eddy, 2013)
As tools seem to mostly focus on protein search, the article introduces nhmmer, a tool that uses profile HMMs for DNA homology search at speeds nearly as fast as blastn with sensitive settings. *nhmmer* is designed to search one or more nucleotide queries against a nucleotide sequence database, and for each query, it searches the target database and outputs a ranked list of the hits with the most significant matches to the query. The hits are assigned a similarity score in bits, along with an E-value indicating the expected number of false positives at a threshold of the score (Supplemental image 1, Supplemental image 2, Supplemental image 3).
nhmmer uses a series of acceleration filters that depend on simpler approximations of the final Forward score of a hit. These filters are based on those used in the HMMER3 protein search tools but have been modified to work in the context of long target sequences. The initial filter scans along the target sequence with a fast un-gapped Viterbi alignment, using a reduced-precision, 16-way vector-parallel approach (Farrar, 2007), which essentially is an optimization technique that allows the processing of 16 calculations simultaneously using a single instruction. Candidate alignments passing this filter then undergo the full rigor of a Forward/Backward alignment to the query.
The tool was tested using the RMARK3 (Nawrocki and Eddy, 2013).
The authors anticipate that *nhmmer* will benefit other domains of DNA sequence comparison that depend on discriminative detection of remote homologs.

HMMRATAC

Several assays exist to identify open chromatin regions, including the popular Assay for Transposase-Accessible Chromatin with high-throughput sequencing (ATAC-Seq) assay. The article (Tarbell and Liu, 2019) introduces a new machine learning method called *HMMRATAC* that identifies open chromatin regions more accurately by decomposing an ATAC-seq dataset into different layers of coverage signals corresponding to nucleosome-free regions (NFRs) or nucleosomal regions and learning the relationships between these signals using a Hidden Markov Model. The algorithm also utilizes the Baum-Welch algorithm to train the Hidden Markov Model on the ATAC-seq data and can be used to evaluate the quality of the data by checking the length distribution of transposition fragments. HMMRATAC is also useful for identifying potential transcription factor binding sites and can be extended to identify differentially accessible regions between two or more conditions.

The comparison of results and effectiveness was done with MACS2 (Gaspar, 2018), F-Seq (Boyle et al., 2008). Analysis done on human GM12878 cell line and human monocyte data from a publicly available database (Kodama et al., 2012).

HMMRATAC outperformed existing methods used for ATAC-seq analysis, including identifying active and/or open chromatin regions and can be integrated into a typical analysis pipeline for ATAC-seq data. (Tarbell and Liu, 2019)
ATAC-seq has advantages over other methods for identifying open chromatin regions, such as low starting material requirement and simple protocol and it outperforms other computational methods that integrate ChIP-seq, DNase-seq and FAIRE-seq. (Yan et al., 2020)

Overview of Tooling

Out of the 4 mentioned tools, 2 of them are readily available on the Galaxy platform. They can be integrated into the existing set of tools for nucleosome position prediction to expand the use cases and provide a more complete workflow that just using the tools on their own. While these tools are available, they are not directly a part of the instance of Galaxy and are not directly usable without additional configuration. The other two tools are publicly available on GitHub but are not a part of the Galaxy platform therefore they require additional wrappers to enable them and then the configuration mentioned to be able to use them in the Galaxy instance for a unified workflow.
These tools will provide multiple choices of prediction possibilities and allow to work with separate implementations of algorithms, as each tool used has its own, as well as different types of data, that could be acquired from databases or uploaded by the user. Such flexibility and integration provide the user with easier access to mentioned tools without doing additional manual work and can not only save time, but provide explanations required to be able to quickly dive into the tool without prior knowledge.

Literature analysis conclusion

Overall, to discover what methods are being used and are readily available, we had to analyse types of hidden Markov models, existing tools, and algorithms. Hidden Markov Models themselves come in a variety of forms, a lot of which can be applied to nucleosome position detection. Such variations have already been used to create working software for nucleosome position detection with varying results. The Galaxy Platform provides a stable base for such tools, that is provided on clusters by the Galaxy Platform itself, as well as the ability to run it on a local machine. Tools can easily be used together with outputs easily transferable, without any programming knowledge, also it provides the tools required to have a complete workflow. While some are readily available for use or integration, others require a compatibility layer – a wrapper – to have the ability to be integrated. In this case, a variety of factors appear and can be taken into consideration when adding and using such tools. Such factors could be:

- Speed of the algorithm

- Accuracy of prediction
- Ease of use
- Type of data used for prediction

Overall, studies show that HMMs can prove useful in nucleosome position prediction, the method has a plethora of tools available for such cases, but it seems to lack a unified workflow, for which the Galaxy Platform provides a great scaffolding to be used together, alongside other required tools. However, this does not imply in any way that the tools are unusable on their own.

METHODS

The 'Methods' section of this thesis outlines the computational approach and techniques employed to predict nucleosome positions. Accurate prediction of these positions is crucial for understanding chromatin organization, gene regulation, and DNA-protein interactions. The goal of each method is to find suitable parameters that provide good results and in the end pick the option that is the most suitable for nucleosome detection.

Data

Data sources and collection

The data used in this study were obtained from publicly available FASTA files, encompassing a diverse range of biological samples. These included apoptotic human cells, CD4+ human cells, stress-resilient mice, stress-susceptible mice, and a control group of mice. Mice sequences were all named, and the human sequences were unnamed, therefore required additional processing. Said sequences were mostly 219 base pairs long, some of them had unfiltered missing bases and different lengths and had to be processed as well. Additionally, we utilized a set of 204 sequences from another study, selecting only the data with a predicted nucleosome centre range of up to 5 base pairs. The FASTA files provided a comprehensive dataset that allowed for the analysis of nucleosome positioning patterns across various cell types and conditions.

The three datasets of mouse brain nucleus accumbens cells are available in Zenodo (https://zenodo.org/record/3813510, accessed 2 May 2023) (Pranckevičienė et al., 2022, Sun et al., 2015). The nucleosome centre is predicted to be in the middle of the sequence in these datasets.
The 204 sequence data available as a supplementary file to the *dnapatterntools* article (Pranckevičienė et al., 2022), https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9102330/bin/ijms-23-04869-s001.zip (accessed 2 May 2023). After filtration, this dataset included sequences from these species: African green monkey, mouse, yeast, simian virus, frog, mouse mammary tumour virus, human, human immunodeficiency virus, rat, fruit fly.
The datasets of apoptotic cells and CD4+ cells were received from the authors of *Nucleosome loss in GC-Rich promoters* article (Hosid and Ioshikhes, 2014)

Dataset characteristics

The dataset was characterized by a wide range of sequence sizes, with the number of sequences varying from around 140,000 to around 700,000 in the primary data sources. This diversity in the dataset allowed for the assessment of nucleosome prediction performance across different scales of data complexity. The smaller set of 204 sequences (Pranckevičienė et al., 2022), with a narrower

predicted nucleosome centre range, served as a complementary dataset to evaluate the specificity of our approach in detecting closely spaced nucleosome positions.

Data pre-processing

Prior to analysis, the raw FASTA sequences were pre-processed to ensure data quality and compatibility with the *HMMER* suite. The pre-processing steps included the removal of low-quality reads, contaminants, and any ambiguous bases. This resulted in a high-quality dataset ready for subsequent analysis with the *HMMER* suite and the custom TypeScript software. To facilitate the use of the pre-processed data in the *HMMER* suite, the custom software tool was employed to convert the cleaned sequences into a format compatible with nhmmer. For additional testing data, when training smaller HMMs, datasets were split up into two pieces – one for training and the other for testing the accuracy of the HMM on similar data. Another way to test was to cross-validate results by training the HMM on one dataset and testing the results on another.

Software

HMMER suite and multiple sequence alignment tools

In this study, we utilized the *HMMER* suite (version 3.3) for sequence analysis, specifically employing nhmmer, hmmbuild, and nhmmscan tools. These tools were essential for the detection of nucleosome positioning patterns in the analysed sequences. Additionally, clustalO (version 1.2.4) (Sievers et al., 2011) and clustalW (version 2.1) (Thompson et al., 1994) were used for multiple sequence alignment, enabling the comparison of sequences and the identification of conserved regions across different samples.

Custom software with NodeJS and TypeScript

The custom software tool employed in this study was developed using TypeScript and ran on NodeJS (v20.1.0). This software was designed to pre-process and manipulate sequence data, facilitating its analysis with the *HMMER* suite and multiple sequence alignment tools. The custom software had several application modes, catering to different aspects of data processing and analysis. The software is compiled locally using the TypeScript compiler. A part of the dependencies and the use of software is done via the Node Package Manager (npm) and the compilation can be done via npm command provided with the software. The instructions on how to run the main functionality of the software are provided alongside it in a public GitHub repository (https://github.com/IndirasM/ts-hmmer, accessed 11 May 2023).

Custom software application modes

The custom software tool offered various application modes, each serving a specific purpose in the data pre-processing and analysis pipeline. These modes included:

- Generator: Produced data by generating 147 bp segments from each sequence, moving the window by 1 bp, and repeating the process.
- Preparator: Cleaned and prepared files for hmmbuild use, with the option to merge multiple sequence files.
- Merge: Combined sequences into one large sequence.
- Lowercase For Alignment: Prepared files for *MAFFT* alignment by converting nucleotides to lowercase.
- Split: Split input files into chunks of 100 sequences, performed multiple sequence alignment with ClustalO, and used hmmbuild to create a series of HMMs for each cluster, ultimately combining them into a single HMM database.
- Sequence Naming: Added indices as names to sequences in FASTA files that lacked names.
- Unique Filter: Filtered out non-unique sequences based on string uniquity.
- Sequence shortener: Extracted a 140 bp section from the middle of each sequence, focusing on the centre of the nucleosome.
- Filtered nhmmer runner: Executed the nhmmer command and output a filtered file with 0-hit outputs removed.

Additional software and tools

In addition to the aforementioned software, we employed *MAFFT* (version 7) (Katoh et al., 2002) to perform multiple sequence alignments on large files. *MAFFT*, running on a cluster, offered an experimental feature for aligning large files without the need for splitting them up, providing an alternative method for handling large datasets. Furthermore, we made use of the Galaxy Platform, specifically the FASTA filtering and merging tools provided by GalaxyProteomics, to assist in data manipulation and pre-processing.

These application modes allowed for the versatile handling of sequence data throughout the analysis process, enhancing the efficiency and accuracy of the computational approach.

Methodology

Data preparation

The methodology employed in this research consisted of several key steps, beginning with data preparation. This initial phase ensured that the sequence data was compatible with the HMMER suite and *MAFFT*. The custom software, as well as Filtering and Merging tools from

GalaxyProteomics were used on a local instance of the Galaxy Platform to prepare the data. To prepare, sequence length was set to be exactly 219 bp and duplicates were to be removed.
The custom software was utilized to pre-process all of the previously mentioned data files. They were matched by the lengths of all sequences, ensuring consistency across the dataset. The custom software automatically figures out the most common length of the sequences in a FASTA file and uses the length to filter out unusable sequences. For the mouse datasets, the most common length was 219 bp while the apoptotic and CD4+ datasets did not require additional filtering for length as all were already 401 bp long. This software also filtered out sequences containing ambiguous 'N' symbols, as these could introduce uncertainty and bias into the analysis. In addition, the custom software was used to name unnamed sequences, namely the apoptotic and the CD4+ datasets.

To further refine the dataset, Filtering and Merging tools available on a local instance of the Galaxy Platform were employed to obtain unique sequences. This step helped eliminate redundancy in the dataset and ensured that the analysis focused on distinct, biologically relevant nucleosome patterns. This filtering was done on the previously processed mouse datasets.

After the cleaning of data, the conversion to lowercase was done on the control, resilient and susceptible mouse data sets to prepare it for multiple sequence analysis using *MAFFT*. With the pre-processing complete, data was aligned on the *MAFFT* cluster using the standard parameters. For the apoptotic dataset, it was not possible to use the entire dataset as *MAFFT* currently accepts up to 700 000 sequences, therefore sequences were removed from the end and stayed with a remainder of 699 997 sequences.

For the 204 sequences dataset, only the ones with the range of nucleosome centre of 5 or less were kept. They were exported as CSV, transferring to an unnamed FASTA format by hand, as there were only 33 sequences left and using the custom software the sequences were named.
These prepared datasets allowed proceeding to the next step and training the HMMs.

Dataset\Number of sequences	Total number of sequences	Filtered number of sequences
Apoptotic	711873	711873
CD4+	581507	581506
Control mouse	188835	105265
Resilient mouse	179246	96855
Susceptible mouse	179402	94523
204-sequences	173	33

Table 1. Sequences per dataset used.

Full dataset alignment HMM with automatic window selection

The first method for predicting nucleosome positions within DNA involved creating HMMs from full alignments of the control, susceptible, and resilient mice datasets. This approach aimed to identify common patterns across the datasets that could be indicative of nucleosome positioning. This was done via using aligned datasets retrieved from the *MAFFT* software and creating a HMM by using the *hmmbuild* command (Supplementary image 4).

With this strategy, both *hmmbuild* and *nhmmer* were used without a set *--w_length* parameter. The HMMER software was allowed to automatically determine the optimal window length for each HMM, based on the input sequences. This approach provided flexibility and allowed the software to adjust the window length according to the sequence data, potentially capturing subtle variations in nucleosome positioning patterns, but with the allowed freedom it can capture details of sequence parts which are outside of the 147 bp nucleosome DNA strand.

The created HMM was used as a query against all the previously mentioned datasets by using the *nhmmer* program via one of the modes of the custom software, which allowed to filter out the results without hits on smaller datasets. *nhmmer* returned the hits that were matches between the HMMs and sequences and shows some sequences that were partially a match, but not enough to climb over the inclusion threshold to be considered a full match. Full matches are scored above an automatically set score inclusion threshold and their sequence start and end positions can be compared with known nucleosome positions, however, some results can be filtered out by the heuristic filters that speed up the search. In such cases the filters can be disabled by adding the *--max* option for maximum sensitivity. In majority of cases only the true matches are important, however some possible ones were investigated.

Full dataset alignment HMM with a hard-set window of 147 bp

In contrast to the first approach, this strategy involved setting a hard window length of 147 bp for all HMMs. This length was chosen because nucleosomes typically wrap around 147 bp of DNA. By using a fixed window length, the HMMs were specifically tailored to capture patterns occurring within this biologically relevant length.

In terms of implementation, the approach is identical to the first approach, but *hmmbuild* and *nhmmer* are both invoked using an additional *--w_length=147* parameter. Setting this parameter essentially tells both the HMM and the search algorithm that the sequences of interest will not be longer than 147 bp. The difference between this approach and the first one is that in this case, all the captured hits should show a pattern within a shorter window and therefore could capture less false positives. The result returned by *nhmmer* is presented in an identical manner to the first approach.

Dataset chunking and HMM database

This method for predicting nucleosome positions within DNA focused on dividing the mouse datasets (control, susceptible, and resilient) into smaller chunks and creating a combined HMM database. This approach aimed to identify more specific patterns within smaller subsets of sequences and explore the potential of a comprehensive HMM database in capturing a broader range of nucleosome positioning patterns.

To start with, the mouse datasets were divided into smaller chunks, each containing 100 sequences. Breaking down the datasets into smaller subsets allowed for a more focused analysis of sequence patterns, capturing localized similarities that might be overlooked in a larger, full-alignment-based approach as well as remove the possibility of adapting the model too strongly to training data. Each of the smaller chunks was aligned separately, and an HMM was trained for every small dataset. Once individual HMMs were trained for each chunk, a combined HMM database was created by incorporating all the individual HMMs. This comprehensive HMM database represented the collective patterns and variations present across all the smaller subsets of sequences.

All datasets, including the control, susceptible, resilient mice, apoptotic human cells, CD4+ human cells, and the 204-sequence data, were subsequently analysed against the combined HMM database. All of the mentioned steps are triggered automatically by using the *Split* mode of the custom software written for the task. It essentially creates all the data step by step – outputs the split sequences, aligned files that are created via ClustalO (or ClustalW if options changed) (Supplementary Image 1) as well as each separate HMM and then connects them into a HMM database. Once a database is present, the *Filtered nhmmer* mode of the software can be run. If possible, the software will try to filter out results that showed no hits and only leave relevant output in the file. By searching for hits in the HMM database, this approach aimed to identify similar sequences across the different datasets, which could yield more hits and provide insights into nucleosome positioning.

This method offered an alternative approach to nucleosome position prediction, focusing on smaller subsets of sequences and a comprehensive HMM database. Comparing the results obtained with the first method, which employed full alignments, allowed for a more in-depth understanding of nucleosome positioning patterns in the analysed datasets as well as provided improved sensitivity.

Smaller HMM database with reserved dataset part for testing

This method for predicting nucleosome positions within DNA involved a similar approach to the second method, with the mouse datasets divided into smaller chunks and individual HMMs trained for each chunk. However, this method incorporated cross-validation by reserving a portion of the data for testing purposes. In contrast to the two other approaches, which were testing the viability of discovering nucleosome positions in general – even on training data - this approach aimed to assess the performance of the HMM database in predicting nucleosome positions on an independent dataset, providing insights into the generalizability of the predictions.

A portion of the data was removed from the large datasets (control, susceptible, and resilient mice) and reserved for use as testing data. This step ensured that the HMM database was trained on a distinct set of sequences, enabling a more accurate evaluation of its predictive performance on unseen data.

Similar to the second method, the remaining mouse datasets were divided into smaller chunks, each containing 100 sequences. This process allowed for a more focused analysis of sequence patterns within the smaller subsets of sequences. Each of the smaller chunks was aligned separately, and an HMM was trained for every small dataset. After training individual HMMs for each chunk, a combined HMM database was created by incorporating all the individual HMMs.

With the HMM database created, the reserved testing data was analysed against it, searching for hits that could predict nucleosome positions. This step provided insights into the performance of the HMM database in predicting nucleosome positions on an independent dataset, gathered by hand from FASTA files, allowing for an evaluation of the generalizability and robustness of the predictions. The third method offered a more rigorous approach to nucleosome position prediction, incorporating cross-validation to assess the predictive performance of the HMM database on unseen data.

Single sequence search against a HMM database with *nhmmscan*

The final method employed for predicting nucleosome positions was the use of *nhmmscan* to test individual sequences against a pre-built HMM database. Unlike *nhmmer*, which is used for searching DNA sequence datasets, *nhmmscan* is designed to search an HMM database with a single query sequence or a small set of sequences. This approach is particularly useful when investigating specific sequences of interest or when a more targeted analysis is required.

nhmmscan was applied to analyze sequences from the filtered 204-sequence dataset using the HMM database created in the second method, which was based on smaller chunks of the mouse datasets (control, susceptible, and resilient mice). By using *nhmmscan*, the research aimed to

identify potential nucleosome positioning patterns from a different angle within the 204-sequence data by comparing them against the HMM database to see if there are models that match this sequence. However – this mode requires *hmmpress* to be used to create database indexing support files.

Low-yield tests

Some testing scenarios seem to provide really low yields of usable results. These are the additional testing scenarios that were tried:

- Unaligned FASTA HMMs
- Maximum sensitivity tests
- Inclusion score and E-value manipulation

The first method attempted was the creation of HMMs using unaligned FASTA sequences. While HMMER is designed to work with aligned sequences, this approach sought to explore whether any meaningful patterns could be identified using unaligned data. Although some rare hits were obtained, they were mostly irrelevant and did not contribute significantly to the nucleosome position predictions. The limited success of this method likely stems from the fact that HMMER relies on aligned sequences to effectively capture conserved patterns and variations across the dataset, which are critical for accurate predictions.

The second method involved adjusting the HMMER software settings to increase its sensitivity, allowing more heuristic filters to be passed. The rationale behind this approach was to identify potential hits that might have been overlooked with the default settings. However, this method rarely yielded any additional hits, and when they did occur, their relevance was uncertain. The lack of significant improvement in the predictions may suggest that the default settings of HMMER already provide a reasonable balance between sensitivity and specificity, and further increasing sensitivity does not necessarily lead to more accurate or meaningful results.

The third method focused on manipulating inclusion scores and E-values in an attempt to identify more hits that could potentially indicate nucleosome positions. By relaxing the inclusion score and E-value thresholds, a greater number of entries were considered as valid. However, this approach seemed to artificially inflate the results, as many of the additional hits had low scores and did not match the expected patterns. This suggests that manipulating inclusion scores and E-values may compromise the specificity and reliability of the predictions, leading to an increased rate of false positives.

In summary, these three additional methods did not yield effective results for predicting nucleosome positions within DNA. The limitations and challenges encountered in these approaches highlight the importance of using well-established methods, such as sequence alignment and appropriate HMMER settings, to ensure accurate and reliable predictions

RESULTS

The data shows that differently trained models have varying degrees of success with the goal being to find the most effective way to create the models as well as on how to search them. One of the main takeaways from the experiments is that when compared between all the methods, data shows that creating a HMM database has the best chance of providing the most accurate hits when searching sequences. One more important point of the results is data shows that hits are lower when comparing between different organisms – this leads to the idea that HMMs are likely to be organism-specific and therefore new models would have to be trained for new organisms. A thing to note would also be that the HMMER suite automatic options for training and searching models are quite good, but some minor changes, such as the length of the read window, could still be required to achieve the best results.

Full dataset alignment HMM with automatic window selection results

The results of the full alignment HMM analysis using various datasets revealed a relatively low number of hits, with many instances in which no hits were detected at all. This outcome can be observed across the different datasets, including the 204-sequence dataset, control, resilient, susceptible, apoptotic, and CD4+ datasets.

W = auto	204	Control	Resilient	Susceptible	Apoptotic	CD4+
Control	0	0	0	0	0	0
Resilient	0	25 (16)	20 (7)	5 (0)	0	0
Susceptible	0	29 (0)	31 (0)	9 (0)	0	0

Table 2. HMMs trained on three full-alignment datasets queried against all the major datasets as well as the 204-sequence dataset. Total hits shown outside the parentheses and the hits that passed the inclusion threshold shown within parentheses. Automatic W length assigned.

When the HMMs were constructed with a window length (W) set to 'auto', the hits for the control dataset yielded no matches across all other datasets. Enabling the maximum sensitivity mode did not provide additional hits. Although there were some hits detected with resilient and susceptible HMMs on the mouse datasets, most of the comparisons resulted in zero hits.

When reviewing the concrete sequences that have passed the inclusion threshold, the only HMM that gave results was the one based on the resilient dataset. In the mouse datasets, the nucleosomal DNA is within the 36-37 and 183-184 base range. Supplementary table 1 shows that of all the sequences were matched with the HMM on Control dataset, only 37,5% of the sequences were within the nucleosome bp range. While the matches on the Resilient dataset were not in the range as shown in Supplementary table 2.

Several factors could contribute to the low hit numbers observed in this analysis. One possibility is that the sequences in the datasets have a high degree of variability, making it difficult for the HMMs to identify conserved patterns and predict nucleosome positions accurately. Additionally, the low hit numbers could also indicate that the HMMs generated from the various datasets might not be comprehensive enough to represent the full range of nucleosome positioning patterns present in the sequences. One more important factor is that multiple sequence alignment for a large amount of sequences introduced large amounts of gaps between bases, which causes the data to become quite bloated.

The models for this are provided alongside the custom software in files named:

- hmms/aligned-control.hmm
- hmms/aligned-resilient.hmm
- hmms/aligned-susceptible.hmm

Full dataset alignment HMM with a hard-set window of 147 bp results

The results of analysis using models with a hard-set window length of 147 bp showed a slight improvement in the number of hits compared to not providing one. However, the overall hit numbers were still relatively low, with many instances of no hits detected across the different datasets, including the 204-sequence dataset, control, resilient, susceptible, apoptotic, and CD4+ datasets.

W = 147	204	Control	Resilient	Susceptible	Apoptotic	CD4+
Control	2 (0)	0	0	0	0	0
Resilient	0	27 (19)	28 (16)	11 (0)	0	0
Susceptible	0	29 (0)	29 (0)	9 (0)	0	0

Table 3. HMMs trained on three full-alignment datasets queried against all the major datasets as well as the 204-sequence dataset. Total hits shown outside the parentheses and the hits that passed the inclusion threshold shown within parentheses. Trained with W length set as 147 bp.

With the W value set, the control dataset yielded two hits with the 204-sequence dataset that did not pass the inclusion threshold, but no matches were found across all other datasets. The resilient and susceptible datasets showed marginally better results, with slightly more hits detected when compared with each other. Apoptotic and CD4+ datasets still had no hits. Despite this improvement, the majority of the comparisons still resulted in zero hits. Interestingly, the two hits on the 204 sequences data were of *Rat* origin. An identical result of 37,5% within nucleosomal bp range was achieved. The reasons why this model is inaccurate could be very similar to why this one is inaccurate as well and even with the small improvements it is not capable of identifying nucleosome positions reliably.

The models for this are provided alongside the custom software in files named:

- hmms/aligned-control-147.hmm
- hmms/aligned-resilient-147.hmm
- hmms/aligned-susceptible-147.hmm

Results with dataset chunking and HMM database results

Three separate HMM databases were used for this part, trained on control, resilient and susceptible datasets. The databases are provided alongside the custom software in files named:

- hmms/hmm_db-controlm.hmm
- hmms/hmm_db-resilientm.hmm
- hmms/hmm_db-susceptiblem.hmm

The results of the analysis using chunked HMMs, and databases demonstrated a significant improvement in the number of hits detected across all datasets compared to the earlier methods. A notable observation from this analysis is that all the models provided hits on every model when run on themselves. Hits were observed when running on human apoptotic and CD4+ datasets as well, which have not shown any hits with previous methods.

	Total HMMs	Control Hits	Resilient Hits	Susceptible Hits	Apoptotic Hits	CD4+ Hits
Control DB	**1053**	**1053**	1044	1045	641	734
Resilient DB	**969**	960	**969**	956	573	670
Susceptible DB	**946**	942	931	**946**	570	651

Table 4. Hit rates of HMMs within the databases. Every model has hits when working with its own dataset.

These results highlight the idea that chunking and using smaller HMMs can prove a lot more responsive in generally detecting similar patterns within sequences, however, this data is a lot more difficult to process due to the fact that every single HMM outputs its own data as if ran by itself, therefore cross-validation between separate HMMs becomes a lot more complex.
For the control database, the largest number of hits for a single HMM was 7323, for resilient DB – 5044, for susceptible – 5033, but within the mouse datasets, each of the FASTA files had around 50 models that had >1000 hits for each of the databases.

	Control hits	Resilient hits	Susceptible hits	Apoptotic hits	CD4+ hits
Control DB	N/A	6360 (5129)	6618 (5244)	444 (31)	338 (34)
Resilient DB	5044 (3684)	N/A	4692 (1997)	525 (318)	699 (376)
Susceptible DB	5033 (3045)	4365 (2660)	N/A	462 (309)	482 (297)

Table 5. Hits of the most sensitive HMM within a database with hits above inclusion threshold within parentheses.

These results seem to show that a lot of results are considered high confidence hits. After reviewing the results, most of them seem to be recognized either as an entirety or a large part of the whole sequence. Within apoptotic sequences of the control database 19 out of 31 sequences have at least partial coverage or are fully within nucleosomal bp range.

For validation of the results, we checked if the hits were within nucleosomal bp range, HMMs with highest hit counts were selected and the percentage of sequences that hit and were within the range were calculated. Results of the database search against the dataset it was trained on was omitted due to it matching lots of entire sequences instead of fitting within a range.

Overall, these results show the capacity of smaller HMMs to capture a lot more sequences that match the patterns, although these numbers are still small when comparing to the number of sequences within the databases, as the numbers within range from hundreds of thousands.

Results with smaller HMM database with reserved dataset part for testing

Results of testing with a reserved dataset are as follow:

	Total HMMs tested	Number of HMMs with results
10 Sequences from Control (Supplementary list 1)	1053	53
10 Sequences from Resilient (Supplementary list 2)	969	9
10 Sequences from Susceptible (Supplementary list 3)	946	34

Table 6. Table of total HMMs in a database and the number of HMMs hit with 10 sequence query files.

When in contrast of the results of the previous method, the numbers become underwhelming. They seemingly show that either the sequences are highly variable and the picked sequences are not suitable for such search, or the models fail to capture details required for new sequences to be properly detected. A more extensive reserved dataset could be tested to determine if these results are consistent across a broader range of sequences, however, this still shows the shortcomings of the model as sequences of similar origin are rarely considered hits within the entire database.

	HMMs with hits	Sequences with hits	Sequences over inclusion threshold	Sequences within nucleosomal bp acceptance window	Percentage of sequences within window
Control	53	71	63	34	54%
Resilient	9	9	7	6	86%
Susceptible	34	37	23	19	83%

Table 7. Table of calculated results for the number of relevant hits. Nucleosomal bp acceptance window includes 1) hit of an almost entire sequence 2) Partial hit on one of the sides that includes at least half of a nucleosomal window 3) A hit entirely within the nucleosomal bp window.

In conclusion, the results of testing with a reserved dataset highlight the challenges associated with predicting nucleosome positions using HMM models. While the hits that were made were over 50% in accuracy, the low amount of hits overall does not show this as an acceptable use of discovering nucleosomes. The underwhelming results suggest the need for further investigation into the causes of these discrepancies, including potentially high variability within the sequences and limitations in the HMM models themselves. According to this result, the main issue with this type of searching is sensitivity, however it is not the heuristic filters that are not passing. By addressing these challenges

and refining the methods employed, it may be possible to improve the accuracy, reliability and sensitivity of nucleosome position predictions.

Single sequence search against a HMM database with *nhmmscan*

Search of HMMs that match the filtered 204 sequence dataset has provided some results, however none of them passed the inclusion threshold and therefore cannot be considered reliable. A few queries had results that have showed up more than once, which might be worth considering.

	Control DB	Resilient DB	Susceptible DB	Total hits
Rat	4	1	2	7
Yeast	1	1	0	2
Human – Seq 1	0	2	0	2
Simian Virus – Seq 1	0	0	2	2
Simian Virus – Seq 2	1	1	0	2
Human – Seq 2	1	1	2	4
Human – Seq 3	1	0	2	3

Table 8. Data of hits that were not within the inclusion threshold but had more than one hit.

All the sequences fit into the bp range for nucleosomes. The rat sequence has picked up the most hits, however, all of them are low confidence and will likely not be a good indicator of nucleosome positioning.

	204 sequence hits
Control Database	7
Resilient Database	9
Susceptible Database	4

Table 9. Number of HMMs within the database that were hit when called with the filtered 204 sequence data.

The results of testing the filtered 204-sequence dataset against the control, resilient, and susceptible HMM databases using *nhmmscan* revealed that some sequences were able to find hits within the databases. This suggests that these sequences contain patterns that match the HMMs in the databases. However, an important observation is that none of the hits passed the inclusion threshold, indicating that the hits might not be strong enough to be high confidence predictions.

There are several possible reasons for the low number of hits passing the inclusion threshold. One potential explanation is that the HMM databases might not adequately represent the data that the

databases were trained on. This implies that patterns likely differ between organisms as the filtered 204 sequence dataset includes only a single sequence of mouse DNA.

Custom software

The custom software developed for this study proved to be a useful tool in facilitating the data preparation and analysis process for predicting nucleosome positions and came through as one of the results of this study. Through its various application modes, it offered a versatile and efficient approach to managing and processing the input data, as well as preparing data for use with other bioinformatics tools such as *HMMER*, *MAFFT*, and the Galaxy Platform.

One of the primary strengths of the custom software was its ability to handle large amounts of data effectively. The modes that are available aided both with the initial investigation of the capabilities of *HMMER* as well as actual investigation and data preparation part. Furthermore, the custom software was adept at performing necessary pre-processing tasks with a streamlined approach. These preprocessing steps were critical to ensuring the consistency of the input data, ultimately providing data that is tailored for this study. The software also can be easily extended and is able to invoke the use of other bioinformatics tools from within.

In conclusion, the custom software significantly contributed to the results of this study by streamlining data preparation and analysis processes, as well as effectively interfacing with other essential bioinformatics tools. However, due to results not showing increased capabilities over other nucleosome detection tools, the decision was not to integrate it with the Galaxy Platform and provide the tool as a standalone entity.

Overview of results

Overall, out of all the methods that have provided results, the search using a database of small HMMs proved the most useful and provided amounts of hits that are a lot higher than any other method. This shows us that the small HMMs are more capable of capturing sequence structure and are less likely to forget the initially learned details. The final approach of using *nhmmscan* did not provide overwhelmingly great results, however, it does allow us to find the most sensitive HMMs easier and might provide insights into the chunk of the dataset that it was trained on.

DISCUSSION

Results of methods

Results have shown that the intended computational prediction of nucleosome positions was not as capable as hoped. In contrast to the other tools available for a similar purpose, this study with the main tool – the *HMMER* suite – has yielded wildly varying results, with the best results being shown by applying HMM databases. However the search was not fruitless and still managed to show capacity of detection with the use of HMM databases showing an affinity for such task.

The initial tests with fully aligned large datasets have only shown rare hits that cannot cover the datasets with useful results. Both with the familiar and unfamiliar datasets the results have been either non-existent or could easily be disregarded due to being in such low numbers. The modification of the read window does have some impact on the results, but overall, it is negligible. However, without the multiple sequence analysis, even if there were hits, data would not be accurate as there would be no clear continuity of the position on where the nucleosome is. This is likely not the best approach as multiple sequence analysis on a large training set can skewer the training data.

The second part of the study yielded results that were better. HMM databases seem to be capable of capturing a lot more details of sequences. Alongside capturing and matching sequences to HMMs, it has shown the ability to completely recognize the training datasets as well as recognize datasets of similar origin with good numbers. Apart from recognizable datasets, it has shown the ability to match some datasets of completely different origin. Even though the databases were trained on mouse sequences, human apoptotic and CD4+ results manage to show hits and therefore provide some insight into possibilities of some similarities between nucleosome position properties within DNA, although the amount of hits is not as high, which can lead us to the notion that the created HMMs are quite organism-specific. It should be kept in mind though, that the results of large HMM databases are not easily manageable and might require additional software.

Further testing with data cut from the same datasets, however, was not performing as well as expected. Even with sequences of similar origin the models seem to have underperformed when comparing to previous test. This could indicate a couple of things: one could be that datasets are highly variable, and it is rather difficult to capture enough details to recognize the majority of sequences well. Another could be that the datasets were too small and included rather obscure variants of sequences.

Testing from a reverse angle with *nhmmscan* – instead of finding sequences that match HMMs, looking for HMMs that match the sequences – also seemed to show rather low yields of hits. This

could either show that HMMs were not as well adapted to multiple kinds of sequences, especially different origin organisms, but it could also indicate that the few hits that were present found particular features that were matching the mouse databases. Another point worth mentioning is that such searching could be a good way to find indications of which HMMs are the most versatile and provide the most information on their own as database results can be difficult to manage.

Custom software

The custom software has played a large role in this study. It was likely possible to use a variety of tools that are available for similar purposes but having everything within one piece of software has helped streamline the research immensely. The choice of programming language might not be the most conventional for biological data, but the platform has shown high adaptability as well as the ability to process data quickly. The software is also easily expandable to have even more entries to streamline the process as well as integrate with other tools that can be ran locally.
There are a few points that could be improved in the future:

- Integration with APIs of software such as *MAFFT*
- Parsing of large amounts of *nhmmer* results and calculating coverage of the entire dataset
- Inclusion of all the necessary tools or improving existing parts, so that there is no need to jump between platforms for filtering data

Integrations would allow to do all of the processes entirely within the confines of the custom software, however multiple sequence alignments with large amounts of data pose challenges, as they can take a lot of time even when running on high-power processing clusters. The ability to parse a large number of results into a more readable format with extracted information could prove useful as well. It could show coverages of data in more depth as well as provide insights into how the models are performing with both familiar and unfamiliar data. As for the inclusion of remaining tools – the filtering part is partially implemented, but it is not completely reliable as of the moment of this study and implementing such features in the future could help with the streamlining of the process.

Discrepancies with existing literature

When considering investigated tools and research, this research did not seem to match the results of other projects, however the results were not fruitless. Created HMMs did not show the capacity to reliably extract features or did with very limited results, but a HMM database shows the possibility that this could be tailored even further as a number of hits have been found. This could be due to several reasons. Such reasons could be that the use of *HMMER* is too tailored for a different purpose and cannot be reliably repurposed. Another reason could be that this particular task requires a different type of HMM that is not provided as is and such significant results as with other projects did not occur. Another issue could be the choice of methods and would require further investigation of

possible methods and comparison with current ones. If work is to be continued with *HMMER* when searching for nucleosomes, then it is likely worth noting that models seem to show the best results when the data is of the same origin as the trained model.

Limitations

The limitations of this research became apparent after seeing the results of performance with data cut from the same datasets. Results seem to show numbers that are too low to consider the tool for extensive use when predicting nucleosome positions, especially when comparing to other methods, such as calculating the position of the nucleosome. The results also show that the use of large multiple sequence alignment file for the creation of a single HMM does not appear to be a good way to predict positions, especially when comparing to a database of smaller HMMs.

Future research

Considering the results of the research, there are a few avenues of possible future research for this topic. One topic could be the calculations for coverage of data to show the performance of specific HMMs or HMM databases on data as there have been varying results. Such tooling could be implemented into the custom software and possibly integrated into other platforms. One more area of interest could be the investigation of reusability of HMM databases on different types of organisms and the investigation of the matching features that were captured. Finally, one more topic could be the integration of machine learning for nucleosome detection as that could possibly expand the feature detection capabilities even more than HMMs.

Out of all the models that were tested within the databases, we have selected the most sensitive model, as the database as a whole does not provide a singular approach to results, and have created a prototype sequence that can be further tested on (Supplementary sequence 1).

CONCLUSIONS

To conclude, the results have shown that with selected tools and data the capabilities of the created model are lacking, but still managed to get results when working with data of the same origin. The model was created and is available for further testing and investigation. From the model with the best results, we've created a prototype sequence that can be further investigated. Alongside it, the results have shown varied success when testing the models created on the data of one organism to map nucleosome positions within another with further investigation being required, although the results were not as good as anticipated when testing with filtered 204 sequence data. HMM databases show a high affinity for detecting nucleosome positions and could likely be tailored even further with more specialized data, but it has to be done keeping in mind that HMMs seem to be quite specific to the organism that it was trained on. Multiple sequence alignment was a necessary step, however it shows better results when it is done in small chunks.

A crucial part of this research was the software developed that allowed us to speed up the research and streamline the entire process. The tool is available as open-source software to use and tailor for custom tasks with documentation available on the basic functions and capabilities of the software, however, the decision was made to not integrate it into galaxy as there is not enough utility as of now to be used alongside different workflows. Despite the discrepancies between the obtained results and existing literature, this research provides a foundation for future studies and advancements in nucleosome position prediction.

RECOMMENDATION

Based on the results of this study, for the purpose of computational mapping of nucleosomes the recommendation to explore and optimize the parameters and methodologies employed can be made. Optimization of parameters such as fine-tuning the *HMMER* suite or other tools could prove crucial. It could also be useful incorporating machine learning techniques to improve predictive accuracy. Secondly, considering the variability in sequences observed, it would be beneficial to expand the scope of datasets used, incorporating a wider range of organisms and cell types to develop more robust and generalizable models but this could be rather difficult as it requires a lot of sequences with confirmed nucleosome positions. Furthermore, as the custom software demonstrated usefulness in streamlining data preparation and analysis, continued development and optimization of such tools are encouraged to enhance their applicability in similar research projects.

REFERENCES

1. Sananbenesi, F., Fischer, A., (2015). Remodeling the susceptibility to stress-induced depression. Nat Med 21, 1125–1126. https://doi.org/10.1038/nm.3970
2. Zhou, X., Blocker, A.W., Airoldi, E.M., O'Shea, E.K., (2016). A computational approach to map nucleosome positions and alternative chromatin states with base pair resolution. eLife 5, e16970. https://doi.org/10.7554/eLife.16970
3. Pranckeviciene, E., Hosid, S., Liang, N., Ioshikhes, I., (2020). Nucleosome positioning sequence patterns as packing or regulatory. PLoS Comput Biol 16, e1007365. https://doi.org/10.1371/journal.pcbi.1007365
4. Onufriev, A.V., Schiessel, H., (2019). The nucleosome: from structure to function through physics. Current Opinion in Structural Biology, Sequences and Topology ● Carbohydrates 56, 119–130. https://doi.org/10.1016/j.sbi.2018.11.003
5. Pranckeviciene, E., Hosid, S., Maziukas, I., Ioshikhes, I., (2022). Galaxy Dnpatterntools for Computational Analysis of Nucleosome Positioning Sequence Patterns. International Journal of Molecular Sciences 23, 4869. https://doi.org/10.3390/ijms23094869
6. Eddy, S.R., (2004). What is a hidden Markov model? Nat Biotechnol 22, 1315–1316. https://doi.org/10.1038/nbt1004-1315
7. Xi, L., Fondufe-Mittendorf, Y., Xia, L., Flatow, J., Widom, J., Wang, J.-P., (2010). Predicting nucleosome positioning using a duration Hidden Markov Model. BMC Bioinformatics 11, 346. https://doi.org/10.1186/1471-2105-11-346
8. Moser, C., Gupta, M., (2012). A Generalized Hidden Markov Model for Determining Sequence-based Predictors of Nucleosome Positioning. Stat Appl Genet Mol Biol 11, 10.2202/1544-6115. https://doi.org/10.2202/1544-6115.1707
9. Peckham, H.E., Thurman, R.E., Fu, Y., Stamatoyannopoulos, J.A., Noble, W.S., Struhl, K., Weng, Z., (2007). Nucleosome positioning signals in genomic DNA. Genome Res 17, 1170–1177. https://doi.org/10.1101/gr.6101007
10. Zhao, Y., Wang, J., Liang, F., Liu, Y., Wang, Q., Zhang, H., Jiang, M., Zhang, Zhewen, Zhao, W., Bao, Y., Zhang, Zhang, Wu, J., Asmann, Y.W., Li, R., Xiao, J., (2019). NucMap: a database of genome-wide nucleosome positioning map across species. Nucleic Acids Research 47, D163–D169. https://doi.org/10.1093/nar/gky980
11. Tsui, K., (2013). Defining Nucleosome Occupancy and Positioning: Evolution and the Role of Trans-acting Factors. https://hdl.handle.net/1807/36020
12. Tarbell, E.D., Liu, T., (2019). HMMRATAC: a Hidden Markov ModeleR for ATAC-seq. Nucleic Acids Research 47, e91. https://doi.org/10.1093/nar/gkz533
13. Fang, K., Li, T., Huang, Y., Jin, V.X., (2021). NucHMM: a method for quantitative modeling of nucleosome organization identifying functional nucleosome states distinctly associated

with splicing potentiality. Genome Biology 22, 250. https://doi.org/10.1186/s13059-021-02465-1

14. Sela, I., Wolf, Y.I., Koonin, E.V., (2018). Genome plasticity, a key factor of evolution in prokaryotes (preprint). Microbiology. https://doi.org/10.1101/357400
15. Shtumpf, M., Piroeva, K.V., Agrawal, S.P., Jacob, D.R., Teif, V.B., (2022). NucPosDB: a database of nucleosome positioning in vivo and nucleosomics of cell-free DNA. Chromosoma 131, 19–28. https://doi.org/10.1007/s00412-021-00766-9
16. Buitrago, D., Codó, L., Illa, R., de Jorge, P., Battistini, F., Flores, O., Bayarri, G., Royo, R., Del Pino, M., Heath, S., Hospital, A., Gelpí, J.L., Heath, I.B., Orozco, M., (2019). Nucleosome Dynamics: a new tool for the dynamic analysis of nucleosome positioning. Nucleic Acids Research 47, 9511–9523. https://doi.org/10.1093/nar/gkz759
17. Durbin, R., Eddy, S.R., Krogh, A., Mitchison, G., (1998). Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids, 1st ed. Cambridge University Press. https://doi.org/10.1017/CBO9780511790492
18. Visser, I., Speekenbrink, M., (2022). Multivariate Hidden Markov Models, in: Visser, I., Speekenbrink, M. (Eds.), Mixture and Hidden Markov Models with R, Use R! Springer International Publishing, Cham, pp. 201–230. https://doi.org/10.1007/978-3-031-01440-6_6
19. Fine, S., Singer, Y., Tishby, N., (1998). The Hierarchical Hidden Markov Model: Analysis and Applications. Machine Learning 32, 41–62. https://doi.org/10.1023/A:1007469218079
20. Galaxy Community, (2022). The Galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2022 update. Nucleic Acids Res 50, W345-351. https://doi.org/10.1093/nar/gkac247
21. Gupta, M., (2007). Generalized Hierarchical Markov Models for the Discovery of Length-Constrained Sequence Features from Genome Tiling Arrays. Biometrics 63, 797–805. https://doi.org/10.1111/j.1541-0420.2007.00760.x
22. Ye, F.X.-F., Ma, Y., Qian, H., (2017). Estimate exponential memory decay in Hidden Markov Model and its applications. https://doi.org/10.48550/arXiv.1710.06078
23. Yuan, G.-C., Liu, J.S., (2008). Genomic Sequence Is Highly Predictive of Local Nucleosome Depletion. PLoS Comput Biol 4, e13. https://doi.org/10.1371/journal.pcbi.0040013
24. Segal, E., Fondufe-Mittendorf, Y., Chen, L., Thåström, A., Field, Y., Moore, I.K., Wang, J.-P.Z., Widom, J., (2006). A genomic code for nucleosome positioning. Nature 442, 772–778. https://doi.org/10.1038/nature04979
25. Yuan, G.-C., Liu, Y.-J., Dion, M.F., Slack, M.D., Wu, L.F., Altschuler, S.J., Rando, O.J., (2005). Genome-Scale Identification of Nucleosome Positions in *S. cerevisiae*. Science 309, 626–630. https://doi.org/10.1126/science.1112178
26. Addison, P.S., (2005). Wavelet transforms and the ECG: a review. Physiol. Meas. 26, R155. https://doi.org/10.1088/0967-3334/26/5/R01

27. Potter, S.C., Luciani, A., Eddy, S.R., Park, Y., Lopez, R., Finn, R.D., (2018). HMMER web server: 2018 update. Nucleic Acids Research 46, W200–W204. https://doi.org/10.1093/nar/gky448
28. Mistry, J., Finn, R.D., Eddy, S.R., Bateman, A., Punta, M., (2013). Challenges in homology search: HMMER3 and convergent evolution of coiled-coil regions. Nucleic Acids Research 41, e121. https://doi.org/10.1093/nar/gkt263
29. Wheeler, T.J., Eddy, S.R., (2013). nhmmer: DNA homology search with profile HMMs. Bioinformatics 29, 2487–2489. https://doi.org/10.1093/bioinformatics/btt403
30. The ENCODE Project Consortium. An integrated encyclopedia of DNA elements in the human genome. *Nature* 489, 57–74 (2012). https://doi.org/10.1038/nature11247
31. Farrar, M., (2007). Striped Smith–Waterman speeds database searches six times over other SIMD implementations. Bioinformatics 23, 156–161. https://doi.org/10.1093/bioinformatics/btl582
32. Nawrocki, E.P., Eddy, S.R., (2013). Infernal 1.1: 100-fold faster RNA homology searches. Bioinformatics 29, 2933–2935. https://doi.org/10.1093/bioinformatics/btt509
33. Gaspar, J.M., (2018). Improved peak-calling with MACS2. https://doi.org/10.1101/496521
34. Boyle, A.P., Guinney, J., Crawford, G.E., Furey, T.S., (2008). F-Seq: a feature density estimator for high-throughput sequence tags. Bioinformatics 24, 2537–2538. https://doi.org/10.1093/bioinformatics/btn480
35. Kodama, Y., Shumway, M., Leinonen, R., on behalf of the International Nucleotide Sequence Database Collaboration, (2012). The sequence read archive: explosive growth of sequencing data. Nucleic Acids Research 40, D54–D56. https://doi.org/10.1093/nar/gkr854
36. Yan, F., Powell, D.R., Curtis, D.J., Wong, N.C., (2020). From reads to insight: a hitchhiker's guide to ATAC-seq data analysis. Genome Biology 21, 22. https://doi.org/10.1186/s13059-020-1929-3
37. Sun, H., Damez-Werno, D.M., Scobie, K.N., Shao, N.-Y., Dias, C., Rabkin, J., Koo, J.W., Korb, E., Bagot, R.C., Ahn, F.H., Cahill, M.E., Labonté, B., Mouzon, E., Heller, E.A., Cates, H., Golden, S.A., Gleason, K., Russo, S.J., Andrews, S., Neve, R., Kennedy, P.J., Maze, I., Dietz, D.M., Allis, C.D., Turecki, G., Varga-Weisz, P., Tamminga, C., Shen, L., Nestler, E.J., (2015). ACF chromatin-remodeling complex mediates stress-induced depressive-like behavior. Nat Med 21, 1146–1153. https://doi.org/10.1038/nm.3939
38. Hosid, S., Ioshikhes, I., (2014). Apoptotic Lymphocytes of H. sapiens Lose Nucleosomes in GC-Rich Promoters. PLOS Computational Biology 10, e1003760. https://doi.org/10.1371/journal.pcbi.1003760
39. Sievers, F., Wilm, A., Dineen, D., Gibson, T.J., Karplus, K., Li, W., Lopez, R., McWilliam, H., Remmert, M., Söding, J., Thompson, J.D., Higgins, D.G., (2011). Fast, scalable generation of high-quality protein multiple sequence alignments using Clustal Omega. Mol Syst Biol 7, 539. https://doi.org/10.1038/msb.2011.75

40. Thompson, J.D., Higgins, D.G., Gibson, T.J., (1994). CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. Nucleic Acids Res 22, 4673–4680.
41. Katoh, K., Misawa, K., Kuma, K., Miyata, T., (2002). MAFFT: a novel method for rapid multiple sequence alignment based on fast Fourier transform. Nucleic Acids Research 30, 3059–3066. https://doi.org/10.1093/nar/gkf436

SUMMARY

This study on DNA properties of computationally mapped nucleosome positions was done by Indiras Maziukas, Vilnius University, for the systems biology study programme, supervised by doc. dr. Erinija Pranckevičienė.

This study aimed to develop a Hidden Markov Model based computational nucleosome mapping tool and integrate it into an existing Galaxy instance, addressing the need for accurate and user-friendly tools in nucleosome position prediction. The HMM-based algorithm was created and trained using known human and mouse nucleosomal sequences and tested on 204 nucleosomal sequences with known nucleosome positions. Custom software was employed for data preparation and analysis, streamlining the workflow with documentation available for its use. The HMM-based tool and additional helper tools were not integrated into the Galaxy Platform, but the platform itself was used. Despite the model's limited accuracy in predicting nucleosome positions when compared to existing literature, valuable insights were gained regarding the challenges and limitations of computational predictions in this field. The study highlights the importance of optimizing parameters and methodologies, expanding the scope of datasets. Overall, this research contributes to the growing body of knowledge in nucleosome position prediction and supports further efforts to understand the complex mechanisms underlying gene regulation and chromatin structure.

SUMMARY IN LITHUANIAN

Skaičiavimais nustatytų nukleosomų pozicijų DNR savybių tyrimą atliko Vilniaus universiteto sistemų biologijos programos studentas Indiras Maziukas, prižiūrėjo doc. dr. Erinija Pranckevičienė.

Šio tyrimo tikslas buvo sukurti paslėptu Markovo modeliu pagrįstą nukleosomų pozicijų radimo įrankį ir integruoti į egzistuojantį „Galaxy“ platformos egzempliorių taip sprendžiant tikslių ir patogių naudoti nukleosomų padėčių prognozavimo įrankių poreikį. Paslėptuoju Markovo modeliu pagrįstas algoritmas buvo sukurtas ir apmokytas naudojant žinomas žmogaus ir pelės nukleosomų turinčias DNR sekas ir išbandytas naudojant 204 sekas su žinomomis nukleosomų pozicijomis. Duomenų paruošimui ir analizei buvo naudojama pritaikyta programinė įranga, kuri supaprastino darbo eigą. Kartu buvo pateikta pritaikytos programinės įrangos dokumentacija. HMM pagrįsta priemonė ir papildomi pagalbiniai įrankiai nebuvo integruoti į "Galaxy" platformą, tačiau pati platforma buvo naudojama duomenų paruošimui.

Lyginant su esama literatūra, modelio tikslumas buvo ribotas, bet nepaisant to gauta vertingų įžvalgų apie šios srities kompiuterinių prognozių iššūkius ir apribojimus. Tyrime pabrėžiama parametrų ir metodikų optimizavimo svarba, plečiant duomenų rinkinių apimtį. Apskritai šis tyrimas prisideda prie augančio žinių kiekio nukleosomų padėties prognozavimo srityje ir padeda toliau stengtis suprasti sudėtingus mechanizmus, kuriais grindžiamas genų reguliavimas ir chromatino struktūra.

APPENDICES

Sequence index	Start range	End range	Within nucleosome range (36-37:183-184) or envelops whole
1	176	55	Yes
2	136	52	Yes
3	122	38	Yes
4	113	26	No
5	100	11	No
6	1	90	No
7	97	211	No
8	105	217	No

Supplementary table 1. Matches within resilient full alignment HMM on Control dataset. Data had duplicate matches due to how HMM learned. Of all sequences matched with the HMM only 37,5% of sequences were within the nucleosome range.

Sequence index	Start range	End range	Within nucleosome range (36-37:183-184) or envelops whole
1	1	90	No
2	142	218	No

Supplementary table 2. Matches within resilient full alignment HMM on Resilient dataset. Data had duplicate matches due to how HMM learned. Of all sequences matched with the HMM, none of sequences were within the nucleosome range.

Supplementary list 1. 10 randomly selected DNA sequences from within the control mouse dataset that were cut out and tested with as the reserved dataset part listed for reproducibility.

Control dataset:

- chr4,118606503,118606503,NM_146399,chr4,118606545,118606545,chr4,118606479,118606578,SRR1138261.32964085,118606545,33.749600,60::chr4:118606459-118606678,41
- chr19,39188387,39188387,NM_001011707,chr19,39188585,39188585,chr19,39188576,39188675,SRR1138263.118304988,39188585,23.104900,60::chr19:39188556-39188775,197
- chr7,51986851,51986851,NM_001289839,chr7,51987165,51987165,chr7,51987091,51987190,SRR1138263.45779761,51987165,35.136100,60::chr7:51987071-51987290,313
- chr10,128548540,128548540,NM_146267,chr10,128548966,128548966,chr10,128548957,128549056,SRR1138261.65653082,128548966,30.678200,60::chr10:128548937-128549156,425
- chr2,147536428,147536428,NR_046014,chr2,147536800,147536800,chr2,147536762,147536861,SRR1138262.18766657,147536800,24.610900,60::chr2:147536742-147536961,371
- chr3,90074070,90074070,NR_105979,chr3,90074105,90074105,chr3,90074012,90074111,SRR1138263.26231029,90074105,33.507800,60::chr3:90073992-90074211,34
- chr9,124036281,124036281,NM_009917,chr9,124036333,124036333,chr9,124036330,124036429,SRR1138263.81552729,124036333,25.022800,60::chr9:124036310-124036529,51
- chr2,103252254,103252254,NM_001145813,chr2,103252390,103252390,chr2,103252293,103252392,SRR1138261.14699530,103252390,33.513700,60::chr2:103252273-103252492,135
- chr3,30135490,30135490,NM_021442,chr3,30135513,30135513,chr3,30135432,30135531,SRR1138262.25336638,30135513,22.527800,60::chr3:30135412-30135631,22
- chr13,119503505,119503505,NM_008002,chr13,119503971,119503971,chr13,119503879,119503978,SRR1138263.98737536,119503971,25.522500,60::chr13:119503859-119504078,465

Supplementary list 2. 10 randomly selected DNA sequences from within the resilient mouse dataset that were cut out and tested with as the reserved dataset part listed for reproducibility.
Resilient dataset:

- chr9,123891781,123891781,NM_007718,chr9,123892693,123892693,chr9,123892594,123892693,SRR1138266.86813920,123892693,28.817500,60::chr9:123892574-123892793,911
- chr4,115557004,115557004,NM_001320545,chr4,115557074,115557074,chr4,115557036,115557135,SRR1138266.36889304,115557074,20.028900,60::chr4:115557016-115557235,69
- chr5,114062363,114062363,NM_001359060,chr5,114062467,114062467,chr5,114062453,114062552,SRR1138265.43730717,114062467,27.328000,60::chr5:114062433-114062652,103
- chr19,4156791,4156791,NM_021485,chr19,4157632,4157632,chr19,4157605,4157704,SRR1138265.121210514,4157632,21.641400,60::chr19:4157585-4157804,840
- chr6,120153827,120153827,NM_198884,chr6,120154736,120154736,chr6,120154695,120154794,SRR1138264.48535273,120154736,26.270100,60::chr6:120154675-120154894,908
- chr2,163552306,163552306,NM_007398,chr2,163552389,163552389,chr2,163552353,163552452,SRR1138264.15880021,163552389,20.178000,60::chr2:163552333-163552552,82
- chr13,61035515,61035515,NM_001145799,chr13,61036370,61036370,chr13,61036369,61036468,SRR1138264.82281809,61036370,26.066800,51::chr13:61036349-61036568,854
- chr7,143523898,143523898,NR_040459,chr7,143524173,143524173,chr7,143524103,143524202,SRR1138266.53570937,143524173,24.322700,60::chr7:143524083-143524302,274
- chr18,35097068,35097068,NM_010481,chr18,35097544,35097544,chr18,35097539,35097638,SRR1138266.121169829,35097544,20.944300,60::chr18:35097519-35097738,475
- chr1,42705044,42705044,NR_027826,chr1,42705914,42705914,chr1,42705817,42705916,SRR1138265.1851497,42705914,20.003000,60::chr1:42705797-42706016,869

Supplementary list 3. 10 randomly selected DNA sequences from within the susceptible mouse dataset that were cut out and tested with as the reserved dataset part listed for reproducibility.

Susceptible dataset:

- chr9,123891781,123891781,NM_007718,chr9,123891903,123891903,chr9,123891851,123891950,SRR1138269.89069291,123891903,38.817100,60::chr9:123891831-123892050,121
- chr2,112399838,112399838,NM_001165935,chr2,112400646,112400646,chr2,112400646,112400745,SRR1138268.15465319,112400646,20.968300,60::chr2:112400626-112400845,807
- chr3,130993408,130993408,NM_027816,chr3,130994173,130994173,chr3,130994161,130994260,SRR1138267.25318998,130994173,24.607200,60::chr3:130994141-130994360,764
- chr15,75755112,75755112,NM_031201,chr15,75755667,75755667,chr15,75755653,75755752,SRR1138267.90291332,75755667,25.230600,60::chr15:75755633-75755852,554
- chr10,3515901,3515901,NM_001304937,chr10,3516563,3516563,chr10,3516464,3516563,SRR1138267.56777218,3516563,20.885100,60::chr10:3516444-3516663,661
- chr14,50857577,50857577,NM_146363,chr14,50858431,50858431,chr14,50858332,50858431,SRR1138268.69559723,50858431,22.807500,60::chr14:50858312-50858531,853
- chr6,66586816,66586816,NM_001166719,chr6,66587085,66587085,chr6,66586991,66587090,SRR1138268.52036602,66587085,23.440500,60::chr6:66586971-66587190,268
- chr2,85193151,85193151,NM_001011534,chr2,85194097,85194097,chr2,85194073,85194172,SRR1138269.14016542,85194097,24.488800,60::chr2:85194053-85194272,945
- chr5,104728305,104728305,NM_008318,chr5,104728330,104728330,chr5,104728329,104728428,SRR1138269.45153205,104728330,28.663100,60::chr5:104728309-104728528,24
- chr17,27259609,27259609,NM_026063,chr17,27260217,27260217,chr17,27260190,27260289,SRR1138269.119026800,27260217,37.006700,60::chr17:27260170-27260389,607

```
>> chr10,7213347,7213347,NR_045946,chr10,7213355,7213355,chr10,7213329,7213428,SRR1138261.60411775,7213355,35.701200,60::chr10:7213309-7213528,7
   score  bias    Evalue   hmmfrom    hmm to     alifrom    ali to      envfrom    env to       sq len      acc
   ------ ----- ---------  -------   -------   --------- ---------   --------- ---------    ---------    ----
 !   25.1  13.1   0.00031         8       150 ..         2       154 ..         1       174 [.        219    0.89
```

Supplementary Image 2. Hit above the inclusion threshold shown in HMMER output. Note the exclamation point in the beginning of the fields.

```
>> chr10,12898443,12898443,NM_181470,chr10,12898991,12898991,chr10,12898893,12898992,SRR1138261.60648273,12898991,31.177400,60::chr10:12898873-12899092,547
   score  bias    Evalue   hmmfrom    hmm to     alifrom    ali to      envfrom    env to       sq len      acc
   ------ ----- ---------  -------   -------   --------- ---------   --------- ---------    ---------    ----
 ?   20.0   3.8     0.012        33       151 ..       101       216 ..        81       219 .]        219    0.89
```

Supplementary Image 3. Hit below the inclusion threshold shown in HMMER output. Note the question mark in the beginning of the fields.

```
Scores for complete hits:
    E-value  score  bias  Sequence                                                                                                                                  start    end  Description
    ------- ------ -----  --------                                                                                                                                  -----  -----  -----------
    1.4e-05   28.7   7.2  chr5,52544562,52544562,NM_001042620,chr5,52545417,52545417,chr5,52545351,52545450,SRR1138263.38976779,52545417,30.982400,60::chr5:52545331-52545550,854      9    215
    2.1e-05   28.1   7.0  chr5,52544562,52544562,NM_001042620,chr5,52545417,52545417,chr5,52545347,52545446,SRR1138263.38976778,52545417,30.982400,60::chr5:52545327-52545546,854      8    218
    7.7e-05   26.3   8.4  chr5,52544562,52544562,NM_001042620,chr5,52545417,52545417,chr5,52545384,52545483,SRR1138263.38976781,52545417,30.982400,60::chr5:52545364-52545583,854     10    183
    0.00027   24.5   4.5  chr5,42178778,42178778,NM_001310601,chr5,42179260,42179260,chr5,42179232,42179331,SRR1138263.38537732,42179260,22.834300,60::chr5:42179212-42179431,481      9    192
    0.00046   23.8   3.1  chr5,42178778,42178778,NM_001310601,chr5,42179193,42179193,chr5,42179192,42179291,SRR1138262.41170317,42179193,20.106200,60::chr5:42179172-42179391,414     39    206
    0.00068   23.2   9.1  chr5,52544562,52544562,NM_001042620,chr5,52545417,52545417,chr5,52545404,52545503,SRR1138263.38976783,52545417,30.982400,60::chr5:52545384-52545603,854      3    163
    0.00081   23.0   3.1  chr5,42178778,42178778,NM_001310601,chr5,42179193,42179193,chr5,42179181,42179280,SRR1138262.41170316,42179193,20.106200,60::chr5:42179161-42179380,414     50    216
    0.00089   22.8   2.9  chr5,42178778,42178778,NM_001310601,chr5,42179260,42179260,chr5,42179180,42179279,SRR1138263.38537729,42179260,22.834300,60::chr5:42179160-42179379,481     51    217
     0.0013   22.3   2.7  chr5,42178778,42178778,NM_001310601,chr5,42179260,42179260,chr5,42179178,42179277,SRR1138263.38537726,42179260,22.834300,60::chr5:42179158-42179377,481     53    218
     0.0055   20.3  10.1  chr5,52544562,52544562,NM_001042620,chr5,52545417,52545417,chr5,52545409,52545508,SRR1138263.38976784,52545417,30.982400,60::chr5:52545389-52545608,854      5    158
     0.0079   19.8   6.7  chr5,42178778,42178778,NM_001310601,chr5,42179260,42179260,chr5,42179253,42179352,SRR1138263.38537734,42179260,22.834300,60::chr5:42179233-42179452,481      7    170
  ------ inclusion threshold ------
       0.01   19.4   1.4  chr5,44566400,44566400,NM_173764,chr5,44566505,44566505,chr5,44566493,44566592,SRR1138261.36588016,44566505,25.555200,60::chr5:44566473-44566692,104     12    193
       0.01   19.4   1.4  chr5,44566400,44566400,NM_173764,chr5,44566505,44566505,chr5,44566501,44566600,SRR1138261.36588017,44566505,25.555200,60::chr5:44566481-44566700,104      4    185
       0.01   19.4   1.4  chr5,44566400,44566400,NM_173764,chr5,44566523,44566523,chr5,44566489,44566588,SRR1138263.38643761,44566523,27.976100,60::chr5:44566469-44566688,122     16    197
       0.01   19.4   1.4  chr5,44566400,44566400,NM_173764,chr5,44566523,44566523,chr5,44566491,44566590,SRR1138263.38643762,44566523,27.976100,60::chr5:44566471-44566690,122     14    195
       0.01   19.4   1.4  chr5,44566400,44566400,NM_173764,chr5,44566523,44566523,chr5,44566494,44566593,SRR1138263.38643763,44566523,27.976100,60::chr5:44566474-44566693,122     11    192
       0.01   19.4   1.4  chr5,44566400,44566400,NM_173764,chr5,44566523,44566523,chr5,44566496,44566595,SRR1138263.38643764,44566523,27.976100,60::chr5:44566476-44566695,122      9    190
      0.011   19.3   1.4  chr5,44566400,44566400,NM_173764,chr5,44566523,44566523,chr5,44566483,44566582,SRR1138263.38643760,44566523,27.976100,60::chr5:44566463-44566682,122     23    203
```

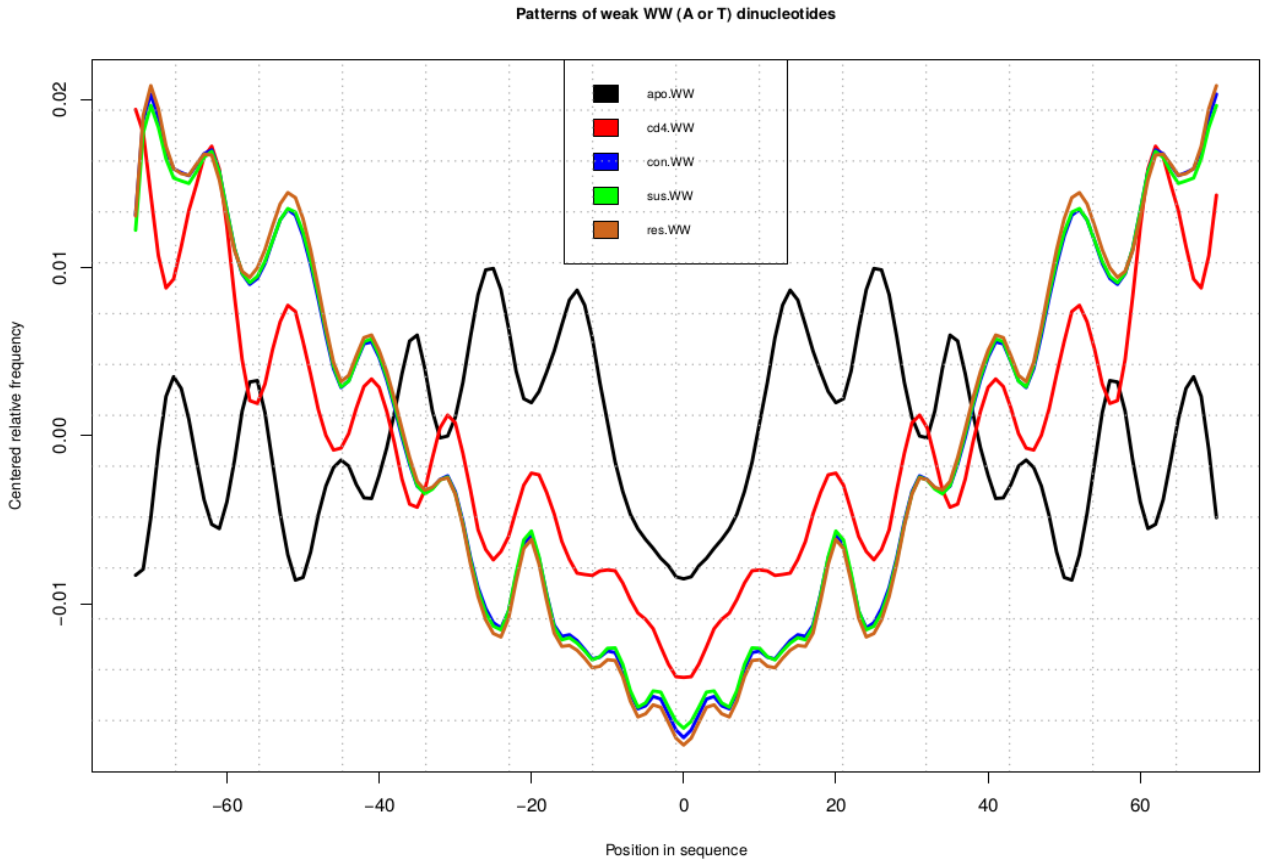
Supplementary Image 4. List of hits between a HMM and sequences.

```
HMMER3/f [3.3 | Nov 2019]
NAME  alignment0
LENG  217
MAXL  369
ALPH  DNA
RF    no
MM    no
CONS  yes
CS    no
MAP   yes
DATE  Sat May 13 01:50:00 2023
NSEQ  100
EFFN  100.000000
CKSUM 2896288451
STATS LOCAL MSV       -10.9222  0.71398
STATS LOCAL VITERBI   -13.0913  0.71398
STATS LOCAL FORWARD    -3.4858  0.71398
HMM          A        C        G        T
            m->m     m->i     m->d     i->m     i->i     d->m     d->d
  COMPO   1.36399  1.46854  1.42962  1.29207
          1.21628  1.52411  1.55492  1.29227
          3.48644  0.72171  0.72676  3.44867  0.03230  0.00000        *
      1   0.96291  1.94469  1.17915  1.78590     69 a - - -
          1.38629  1.38629  1.38629  1.38629
          0.00379  6.27078  6.27078  1.46634  0.26236  4.62126  0.00989
      2   0.75942  1.98895  1.35726  1.98152     70 a - - -
          1.38629  1.38629  1.38629  1.38629
          0.00379  6.27078  6.27078  1.46634  0.26236  2.17777  0.12024
      3   2.22239  2.11494  1.80165  0.50090     71 t - - -
          1.69367  1.37984  1.62099  1.00284
          3.36980  0.10684  2.70408  2.28419  0.10743  4.51205  0.01104
      4   1.10045  2.15656  0.89896  1.93402     86 g - - -
          1.38629  1.38629  1.38629  1.38629
          0.06490  6.29810  2.79696  1.46634  0.26236  2.19942  0.11751
      5   1.27986  4.00466  0.93466  1.16805     87 g - - -
          1.38629  1.38629  1.38629  1.38629
          0.00357  6.32989  6.32989  1.46634  0.26236  2.96592  0.05289
```

Supplementary Image 5. A part of the HMM created via hmmbuild.

```
>chr10,3515901,3515901,NM_001304937,chr10,3516563,3516563,chr10,3516464,3516563,SRR1138267.56777218,3516563,20.885100,60::chr10:
-------------------------CATAACACACAGTGATGATGA---------GGAC
CGGCATGATGAAGGCGAAGA---TGAAGACACAGATTTT-----------G--AGCAGGT
TCTCCCAGTACCATG-------TGGGATGAGA-GAACGTGAGGGTGCAATCT---ATG--
------------G-ACCCTGGAAGGAA-AGAAGTTTCAAGAAAAATTA--GCAGCGACAT
TTAGTTATA---T--CATC--------AG-GTTTTAAAATTGCACCTTTGCCTCCTTTGC
CTAGGTTGTTG-CT----------------------------------------------
--------------------------------------------------------------
---------------
>chr10,3515901,3515901,NM_001304937,chr10,3516563,3516563,chr10,3516468,3516567,SRR1138267.56777219,3516563,20.885100,60::chr10:
-----------------------------ACACACAGTGATGATGA---------GGAC
CGGCATGATGAAGGCGAAGA---TGAAGACACAGATTTT-----------G--AGCAGGT
TCTCCCAGTACCATG-------TGGGATGAGA-GAACGTGAGGGTGCAATCT---ATG--
------------G-ACCCTGGAAGGAA-AGAAGTTTCAAGAAAAATTA--GCAGCGACAT
TTAGTTATA---T--CATC--------AG-GTTTTAAAATTGCACCTTTGCCTCCTTTGC
CTAGGTTGTTG-CTTACA------------------------------------------
--------------------------------------------------------------
---------------
>chr10,3515901,3515901,NM_001304937,chr10,3516563,3516563,chr10,3516480,3516579,SRR1138267.56777220,3516563,20.885100,60::chr10:
-----------------------------------------GATGA---------GGAC
CGGCATGATGAAGGCGAAGA---TGAAGACACAGATTTT-----------G--AGCAGGT
TCTCCCAGTACCATG-------TGGGATGAGA-GAACGTGAGGGTGCAATCT---ATG--
------------G-ACCCTGGAAGGAA-AGAAGTTTCAAGAAAAATTA--GCAGCGACAT
TTAGTTATA---T--CATC--------AG-GTTTTAAAATTGCACCTTTGCCTCCTTTGC
CTAGGTTGTTG-CTTACAGATGGGTAAGAA-----------------------------
--------------------------------------------------------------
---------------
```

Supplementary Image 6. Excerpt from a MSA file after alignment.



Supplementary Figure 1. Representation of patterns as peaks. (Pranckevičienė et al., 2020)

Supplementary sequence 1. This sequence was the output of the best performing model from all the databases. This was the result of the database created from the control mouse dataset.

>alignment141-sample1
CCAGCAATGGAGATGATATAATTATATACGCATCACACAAACTGCATATTCTTACAGCTA
ATATTTTTGAAACGAAGTGGTTCCGAAATCTGCTCCCTTTGACATTGACTTACTACTATT
TTGATACATGAATTCTGTGGCATTACTGACCTTAGGATGCCGCGCACAAGACTATAAGGC
TGCTCCTTGAAGGACTACCTTTTTATATAATAGATGGCTATTGTTCCAGACCGTTTCTGT
CGATAAAGGCATGAATTTAGTGGGCCTAACTATCATTGTCATATATTAGTATACAAGGCG
CACAAGTACAACATGATAACGAAAGTGGATTCTGCTATTCGATAGAGCCGATGAGCCGCC
CTTGACAGGACTCTGATCTCTAAGAGAGCACAAGATAAAACTGTAGGAGGTATCTACATA
TCCAAAAGCTGATGATAACAACGCCAATAAGAGTACTGTTGTACGCAACCAGAGTACCAA
CAATTATTGTCCTAGTACAGGCCACGTAAAATACTGGGAACAAGTCACACATGTAAAGGT
CTATCACGTTGAGGACACACGCTCGAACGGGCTCATCAAAGCCGAGAAGATCTGGAGAAC
AGAATGCCAAAACCTCTACCCAAGACAATTGCAACAAACACGCAGCACTAGCGGTGGTTC
CAAGGATGGTGCAATGGCTTGCAAATGTCGAGGTATCGCTCATAGACCATAACTGTTTGA
ACAAAGGTCTCTGCATGTAGTGGGCTGCAAAGAGGCCCTAAAAAAAGTGGAATCAGGCTC
AGTTAACGGAGATGGTCTTCTTATCCCATCGAAAGTCTTAGTAACTCGGATGAACTCGGA
ACGGATTGAACAATCTCCAGAATTGACATAGTGCTGAGGCATTACACCGGAGAATTCTAA
GAAGGGAGTCATATCCGTTAGTGCAATGAGAAGTGCACCAACTTTGTGACAATGTAGTTT
CCGAAAATTCACAAGACAAATAATGCTTTTATAAAAGCAGTAACTTGTGCAAGGTCCGAA
ACAATGGATTCCTATTGTAGCCATGTCCCATTAGTGTAGGACCGTCCTTTGAATGATGCG
AGCTTACACTGGCGGTCTGTGGCTTACATTCCATAGCAACTTGTCACCCTAGCTTTTCTA
AGGGTGATTAGGAAGTAGGCATGCAG