Orthopoxvirus Genome Analysis for Species Classification and Virulence Prediction

G Gnanashree
BL.SC.U4AIE24014
CSE-AIE, Amrita School of Engineering
Amrita Vishwa Vidyapeetham, Bengaluru
Email: gnanashreegekkala@gmail.com

PVS Lalitha
BL.SC.U4AIE24031
CSE-AIE, Amrita School of Engineering,
Bengaluru
Email: wby1606@gmail.com

Y Likhitha Sree
BL.SC.U4AIE24056
CSE-AIE, Amrita School of Engineering,
Bengaluru
Email: likhithayarraguntla@gmail.com

Abstract

Orthopoxviruses are closely related double-stranded DNA viruses that include Variola, Monkeypox, Cowpox, and Vaccinia viruses. Although these viruses share high genomic similarity, they differ in host range and virulence. Understanding these differences at the genomic level is essential for evolutionary analysis and public health monitoring.

In this study, a systematic genome-based computational analysis was performed on four Orthopoxvirus species. Complete genome sequences were retrieved and curated to construct a high-quality, non-redundant dataset using a k-mer–based cosine similarity approach. Sequence-derived features including nucleotide composition, k-mer frequencies, and dinucleotide bias were extracted to quantitatively represent genomic variation.

Two machine learning models were developed: a multi-class classification model for species identification, and a virulence prediction model to categorize genomes based on pathogenic potential. Feature importance analysis was conducted to determine the genomic attributes contributing most significantly to species differentiation and virulence variation. The results demonstrate that alignment-free genomic features combined with supervised learning provide reliable classification and meaningful biological insights.

Keywords

Orthopoxvirus, Genome Analysis, k-mer Analysis, Machine Learning, Species Classification, Virulence Prediction

I. INTRODUCTION

Orthopoxviruses are large double-stranded DNA viruses belonging to the Poxviridae family and include medically significant species such as Variola virus, Monkeypox virus, Cowpox virus, and Vaccinia virus. Despite sharing substantial genomic similarity, these viruses differ in host specificity, transmission patterns, and virulence. Studying their genomic variations is important for understanding viral evolution, monitoring emerging outbreaks, and supporting vaccine and therapeutic research.

Genome-based analysis provides a robust approach for distinguishing closely related viral species. Traditional alignment-based sequence comparison methods can become computationally intensive when handling complete genomes and large datasets. In contrast, alignment-free approaches such as k-mer analysis enable efficient and scalable genome comparison while preserving informative sequence patterns.

Recent advances in bioinformatics and machine learning have enabled automated feature extraction and predictive modelling from genomic data. However, accurate modelling depends on well-curated, non-redundant datasets and biologically meaningful feature representation.

In this study, a curated dataset is constructed of complete genomes from four Orthopoxvirus species and extract sequence-based features

including nucleotide composition, k-mer frequencies, and dinucleotide bias. Using these features, two supervised machine learning models are developed: one for multi-class species classification and another for virulence prediction. Additionally, feature importance analysis is performed to identify genomic determinants that contribute to species differentiation and variations in pathogenic potential.

II. METHODOLOGY

A. Data Collection

For this project, genome sequences of four Orthopoxvirus species — Variola, Monkeypox, Cowpox, and Vaccinia — were retrieved from the NCBI GenBank database in FASTA format.

The sequences were retrieved using a Python script with the Biopython library (Entrez module), which automated querying the NCBI database and downloading sequences in FASTA format. A total of 250 sequences per virus were collected, resulting in 1,000 sequences overall. Both complete and partial nucleotide sequences were included to capture the diversity of strains, covering multiple years and geographic regions.

As a result of this process, the sequences were successfully collected in a structured and organized dataset, ready for further analysis

B. Data Curation

To ensure a non-redundant dataset, we used a k-mer based cosine similarity method, which is a commonly used approach in bioinformatics for identifying highly similar nucleotide sequences. This method converts each sequence into a vector of k-mer counts — short sub sequences of length k — allowing sequences to be compared numerically rather than by direct alignment.

In this study, k = 6 was chosen. The cosine similarity between each new sequence and previously kept sequences is calculated, and sequences with similarity above a predefined threshold of 0.992 (for Variola, Cowpox, and Vaccinia) are considered redundant and excluded. Monkeypox sequences were treated differently: all sequences were initially kept due to the low diversity observed in preliminary analyses. The program reads sequences from FASTA files using Biopython's SeqIO module, converts sequences to vectors, performs similarity comparisons, and writes the nonredundant sequences back to new FASTA files for each virus.

C. Dataset Specificity Validation

To confirm that the curated sequences were specific to each virus, a few representative sequences from each dataset were checked using NCBI BLASTn. BLAST compares a query sequence against a database to find similar sequences, allowing us to verify that the sequences matched primarily with the intended virus.

For this validation, five sequences per virus were selected from the non-redundant dataset to cover different strains and geographic regions. These sequences were submitted as queries in FASTA format, and the BLAST results were analyzed to ensure that the top hits corresponded to the target virus.

D. Feature Extraction

After completing dataset curation and validation, the next step was to extract informative features from each complete Orthopoxvirus genome so that meaningful comparisons could be made across species. Because the study aims to classify species and predict virulence categories, the feature extraction process was designed to capture characteristics of the genome that reflect evolutionary differences and potential functional variation.

- *Mononucleotide Composition*
  Mononucleotide composition refers to the relative proportion of A,T,G,C within each complete genome. For every genome sequence, the total occurrences of each nucleotide were counted and normalized by the genome length to obtain percentage values.
  These features provide a basic structural profile of each genome. Variations in nucleotide composition may reflect evolutionary divergence and host adaptation patterns, which can assist the ML models in distinguishing between different viral species.
- *GC Content and Strand Asymmetry*
  GC content was calculated as the proportion of guanine and cytosine bases relative to total genome length. GC skew

and AT skew were computed using strand asymmetry formulas.
GC content relates to genome stability and mutation tendencies, while skew values reflect strand-level bias. Together, these metrics add additional discriminatory information and help capture evolutionary differences across species.

- *Dinucleotide Frequencies*
  All dinucleotide combinations were quantified using overlapping two-base windows across each genome and normalized by sequence length.
  Dinucleotide frequencies capture short-range nucleotide dependencies beyond single-base composition. These patterns often vary across viral species and may reflect adaptation to host immune pressure, making them useful for classification and virulence analysis.
- *Observed-to-Expected Ratios*
  For each dinucleotide, an observed-to-expected ratio was calculated using mononucleotide-based expectations.
  This measure highlights enrichment or suppression of specific dinucleotides.
  Such suppression patterns are linked to immune recognition and provide biologically relevant signals for modeling virulence.
- *Trinucleotide (3-mer) Frequencies*
  All trinucleotide combinations were enumerated using a sliding window approach. Since trinucleotides correspond to codons, their distribution reflects codon usage patterns.
  Codon bias can indicate host adaptation and translational efficiency. Differences in these patterns assist in distinguishing species and may relate to differences in viral fitness.

- *Tetranucleotide (4-mer) Frequencies*
  All four-base combinations were calculated using overlapping windows across the genome.
  4-mers capture more specific local motifs beyond codon-level information. These features improve resolution when distinguishing closely related viral genomes.
- *k-mer Entropy*
  Entropy was also computed at the k-mer level to assess motif diversity.
  This metric captures local sequence complexity and complements basic compositional features.
- *Genome Length*
  The total nucleotide length of each complete genome was recorded.
  Genome length differences may arise from insertions or deletions and contribute to structural variation between species.
- *Repeat Density*
  Repetitive regions were identified by detecting recurring sequence patterns and estimating their proportional representation.
  Repeat density reflects aspects of genome organization and may vary across Orthopoxvirus species.

- **BERT Model**
  To complement compositional features, a transformer-based model, DNABERT, was used to generate contextual genome representations. DNABERT is an adaptation of the BERT architecture for DNA sequences, where overlapping k-mers are treated as tokens.In this study, complete Orthopoxvirus genomes were segmented into overlapping k-mers and provided as input to the pretrained DNABERT model. Through self-attention layers, the model captures contextual relationships and long-range dependencies within the genome. The final hidden layer embeddings were aggregated using mean pooling to produce a fixed-length vector (e.g., 768 dimensions) representing each genome.Unlike frequency-based features, these embeddings encode deeper sequence patterns that may help distinguish closely related Orthopoxvirus species and capture subtle genomic signals associated with virulence

variation. The resulting embedding vectors were used as input to both the species classification and virulence prediction models.

III. IMPLEMENTATION DETAILS

A. Dataset Retrieval

```python
from Bio import Entrez, SeqIO

Entrez.email = "wby1606@gmail.com"

viruses = [
    "Variola virus",
    "Monkeypox virus",
    "Cowpox virus",
    "Vaccinia virus"
]

sequences_per_virus = 250

for virus in viruses:
    print(f"\nSearching for {virus}...")

    search_handle = Entrez.esearch(
        db="nucleotide",
        term=f"{virus}[Organism]",
        retmax=sequences_per_virus
    )
    search_results = Entrez.read(search_handle)
    ids = search_results["IdList"]

    print(f"Found {len(ids)} sequences")

    if len(ids) == 0:
        print("No sequences found, skipping.")
        continue

    fetch_handle = Entrez.efetch(
        db="nucleotide",
        id=ids,
        rettype="fasta",
        retmode="text"
    )

    sequences = list(SeqIO.parse(fetch_handle, "fasta"))

    filename = virus.replace(" ", "_") + ".fasta"
    SeqIO.write(sequences, filename, "fasta")

    print(f"Saved to {filename}")
```

B. Data Curation

```python
from Bio import SeqIO
import numpy as np
from itertools import product

input_files = {
    "variola": "Orthopox_dataset/Variola_virus.fasta",
    "monkeypox": "Orthopox_dataset/Monkeypox_virus.fasta",
    "cowpox": "Orthopox_dataset/Cowpox_virus.fasta",
    "vaccinia": "Orthopox_dataset/Vaccinia_virus.fasta"
}

k = 6
thresholds = {
    "variola": 0.992,
    "cowpox": 0.992,
    "vaccinia": 0.992
}

def generate_kmers(k):
    return [''.join(p) for p in product('ATGC', repeat=k)]

all_kmers = generate_kmers(k)
kmer_index = {kmer: idx for idx, kmer in enumerate(all_kmers)}


def seq_to_vector(seq):
    vector = np.zeros(len(all_kmers), dtype=np.float32)
    for i in range(len(seq) - k + 1):
        kmer = seq[i:i+k]
        if kmer in kmer_index:
            vector[kmer_index[kmer]] += 1
    norm = np.linalg.norm(vector)
    if norm > 0:
        vector = vector / norm
    return vector

for virus_name, file_path in input_files.items():
    records = list(SeqIO.parse(file_path, "fasta"))

    if virus_name.lower() == "monkeypox":
        kept_records = records
        removed_count = 0
        non_redundant_count = len(records)
    else:
        kept_records = []
        genome_vectors = []
        threshold = thresholds[virus_name]

        for record in records:
            seq = str(record.seq).upper()
            vec = seq_to_vector(seq)

            is_redundant = False
            for existing_vec in genome_vectors:
                sim = np.dot(vec, existing_vec)
                if sim >= threshold:
                    is_redundant = True
                    break

            if not is_redundant:
                kept_records.append(record)
                genome_vectors.append(vec)

        non_redundant_count = len(kept_records)
        removed_count = len(records) - non_redundant_count

    print("------", virus_name.upper(), "------")
    print("Original sequences      :", len(records))
    print("Removed as redundant    :", removed_count)
    print("Non-redundant obtained  :", non_redundant_count)
    print()

    output_file = virus_name + "_final.fasta"
    SeqIO.write(kept_records, output_file, "fasta")
```

C. Dataset Specificity Validation

```python
from Bio import SeqIO
import numpy as np
from sklearn.metrics.pairwise import cosine_similarity
from itertools import product
import os

input_files = {
    "variola": "variola_virus.fasta",
    "monkeypox": "monkeypox_virus.fasta",
    "cowpox": "cowpox_virus.fasta",
    "vaccinia": "vaccinia_virus.fasta"
}

k = 3
similarity_threshold = 0.98

def generate_kmers(k):
    return [''.join(p) for p in product('ATGC', repeat=k)]

all_kmers = generate_kmers(k)
kmer_index = {kmer: idx for idx, kmer in enumerate(all_kmers)}

def seq_to_vector(seq):
    vector = np.zeros(len(all_kmers))
    for i in range(len(seq) - k + 1):
        kmer = seq[i:i+k]
        if kmer in kmer_index:
            vector[kmer_index[kmer]] += 1
    return vector

for virus_name, file_path in input_files.items():

    records = list(SeqIO.parse(file_path, "fasta"))
    vectors = []
    kept_records = []

    for record in records:
        vec = seq_to_vector(str(record.seq).upper())

        if len(vectors) == 0:
            vectors.append(vec)
            kept_records.append(record)
        else:
            sims = cosine_similarity([vec], vectors)[0]
            if max(sims) < similarity_threshold:
                vectors.append(vec)
                kept_records.append(record)

    print("------", virus_name.upper(), "------")
    print("Original:", len(records))
    print("Non-redundant:", len(kept_records))
    print()

    output_file = virus_name + "_nonredundant.fasta"
    SeqIO.write(kept_records, output_file, "fasta")
```

IV. RESULTS AND ANALYSIS

A. Data Retrieval Results

```
Searching for Variola virus...
Found 250 sequences
Saved to Variola_virus.fasta

Searching for Monkeypox virus...
Found 250 sequences
Saved to Monkeypox_virus.fasta

Searching for Cowpox virus...
Found 250 sequences
Saved to Cowpox_virus.fasta

Searching for Vaccinia virus...
Found 250 sequences
Saved to Vaccinia_virus.fasta
```

B. Dataset Curation Results

```
------ VARIOLA ------
Original sequences      : 250
Removed as redundant    : 184
Non-redundant obtained  : 66

------ MONKEYPOX ------
Original sequences      : 250
Removed as redundant    : 0
Non-redundant obtained  : 250

------ COWPOX ------
Original sequences      : 250
Removed as redundant    : 167
Non-redundant obtained  : 83

------ VACCINIA ------
Original sequences      : 250
Removed as redundant    : 130
Non-redundant obtained  : 120
```

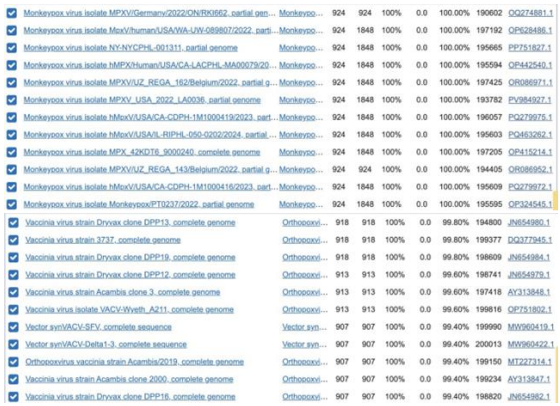
The redundancy removal showed differences between the viruses. Variola had many similar sequences, leaving only a small number of non-redundant sequences. In contrast, all Monkeypox sequences were retained, as preliminary analysis showed low redundancy and high diversity among strains.

Cowpox and Vaccinia displayed moderate redundancy, with a substantial number of unique sequences remaining.

This process effectively reduced duplication while maintaining sequence diversity, resulting in a curated dataset suitable for further analyses.

C. Dataset Specificity Validation Results

Description	Scientific Name	Max Score	Total Score	Query Cover	E value	Per. Ident	Acc. Len	Accession
Variola virus isolate VARV_AFRICA_1964_LUK, complete genome	Variola virus	200	200	100%	3e-47	100.00%	186197	PP405584.1
Taterapox virus, complete genome	Taterapox ...	200	200	100%	3e-47	100.00%	198050	NC_008291.1
Variola virus strain Somalia 1977, complete genome	Variola virus	200	200	100%	3e-47	100.00%	186231	DQ437590.1
Variola virus strain China Horn 1948, complete genome	Variola virus	200	200	100%	3e-47	100.00%	186668	DQ437582.1
Camelpox virus CMS, complete genome	Camelpox ...	200	200	100%	3e-47	100.00%	202205	AY009089.1
Orthopoxvirus camelpox strain Camelpox virus/KZ/Beineu/2023, comp...	Orthopoxvi...	200	200	100%	3e-47	100.00%	202273	PV920573.1
Variola virus strain Pakistan 1969 (Rafiq Lahore), complete sequence	Variola virus	200	200	100%	3e-47	100.00%	185865	DQ437589.1
Variola virus strain Ethiopia 1972 (Eth17.R14-1X-72 Addis), complete ...	Variola virus	200	200	100%	3e-47	100.00%	186648	DQ441425.1
Variola virus strain India 1953 (New Delhi), complete genome	Variola virus	200	200	100%	3e-47	100.00%	186662	DQ441428.1
Camelpox virus strain D1795/20, complete genome	Camelpox ...	200	200	100%	3e-47	100.00%	201398	MZ300858.1
Variola virus isolate VARV_64-7, complete genome	Variola virus	200	200	100%	3e-47	100.00%	186216	PP405594.1
Cowpox virus isolate Brighton_Red-19560227, complete genome	Cowpox vi...	920	5849	100%	0.0	100.00%	221599	PX585982.1
Cowpox virus, complete genome	Cowpox vi...	835	12975	94%	0.0	98.33%	224499	NC_003663.2
Cowpox virus CPXV/Bos taurus/Basrah/BASRAH1/2024 DNA, compl...	Cowpox vi...	835	12948	94%	0.0	98.33%	224499	LC895885.1
Mutant Cowpox virus strain recombinant Brighton Red clone deltaCP...	Cowpox vi...	835	12975	94%	0.0	98.33%	226090	ON549927.1
Cowpox virus BASRAH2 DNA, complete genome	Cowpox vi...	835	12742	94%	0.0	98.33%	224499	LC909709.1
Cowpox virus BASRAHBUFA1 DNA, complete genome	Cowpox vi...	835	12874	94%	0.0	98.33%	224499	LC909707.1
Cowpox virus BASRAHBUFA2 DNA, complete genome	Cowpox vi...	835	12874	94%	0.0	98.33%	224499	LC909708.1
Cowpox virus isolate CPXV_K4207, complete genome	Cowpox vi...	830	12664	94%	0.0	98.12%	224139	KY549150.2
Cowpox virus isolate CPXV_Catpox3L97, complete genome	Cowpox vi...	819	10743	100%	0.0	97.70%	216799	KY549143.2
Cowpox virus isolate CPXV_Catpox5wv1, complete genome	Cowpox vi...	817	13084	100%	0.0	97.51%	224952	KY549144.1
Cowpox virus isolate CPXV_1639, complete genome	Cowpox vi...	815	12925	100%	0.0	97.50%	224936	KY549148.1



Monkeypox virus isolate MPXV/Germany/2022/ON/RKI662, partial gen...	Monkeypo...	924	924	100%	0.0	100.00%	190602	OQ274881.1
Monkeypox virus isolate MoxV/human/USA/WA-UW-089807/2022, parti...	Monkeypo...	924	1848	100%	0.0	100.00%	197192	OP628486.1
Monkeypox virus isolate NY-NYCPHL-001311, partial genome	Monkeypo...	924	1848	100%	0.0	100.00%	195665	PP751827.1
Monkeypox virus isolate hMPX/Human/USA/CA-LACPHL-MA00079/20...	Monkeypo...	924	1848	100%	0.0	100.00%	195594	OP442540.1
Monkeypox virus isolate MPXV/UZ_REGA_162/Belgium/2022, partial g...	Monkeypo...	924	1848	100%	0.0	100.00%	197425	OR086971.1
Monkeypox virus isolate MPXV_USA_2022_LA0036, partial genome	Monkeypo...	924	1848	100%	0.0	100.00%	193782	PV984927.1
Monkeypox virus isolate hMoxV/USA/CA-CDPH-1M1000419/2023, part...	Monkeypo...	924	1848	100%	0.0	100.00%	196057	PQ279975.1
Monkeypox virus isolate hMpxV/USA/IL-RIPHL-050-0202/2024, partial ...	Monkeypo...	924	1848	100%	0.0	100.00%	195603	PQ463262.1
Monkeypox virus isolate MPX_42KDT6_9000240, complete genome	Monkeypo...	924	1848	100%	0.0	100.00%	197205	OP415214.1
Monkeypox virus isolate MPXV/UZ_REGA_143/Belgium/2022, partial g...	Monkeypo...	924	924	100%	0.0	100.00%	194405	OR086952.1
Monkeypox virus isolate hMoxV/USA/CA-CDPH-1M1000416/2023, part...	Monkeypo...	924	1848	100%	0.0	100.00%	195609	PQ279972.1
Monkeypox virus isolate Monkeypox/PT0237/2022, partial genome	Monkeypo...	924	1848	100%	0.0	100.00%	195595	OP324545.1
Vaccinia virus strain Dryvax clone DPP13, complete genome	Orthopoxvi...	918	918	100%	0.0	99.80%	194800	JN654980.1
Vaccinia virus strain 3737, complete genome	Orthopoxvi...	918	918	100%	0.0	99.80%	199377	DQ377945.1
Vaccinia virus strain Dryvax clone DPP19, complete genome	Orthopoxvi...	918	918	100%	0.0	99.80%	198609	JN654984.1
Vaccinia virus strain Dryvax clone DPP12, complete genome	Orthopoxvi...	913	913	100%	0.0	99.60%	198741	JN654979.1
Vaccinia virus strain Acambis clone 3, complete genome	Orthopoxvi...	913	913	100%	0.0	99.60%	197418	AY313848.1
Vaccinia virus isolate VACV-Wyeth_A211, complete genome	Orthopoxvi...	913	913	100%	0.0	99.60%	199816	OP751802.1
Vector synVACV-SFV, complete sequence	Vector syn...	907	907	100%	0.0	99.40%	199990	MW960419.1
Vector synVACV-Delta1-3, complete sequence	Vector syn...	907	907	100%	0.0	99.40%	200013	MW960422.1
Orthopoxvirus vaccinia strain Acambis/2019, complete genome	Orthopoxvi...	907	907	100%	0.0	99.40%	199150	MT227314.1
Vaccinia virus strain Acambis clone 2000, complete genome	Orthopoxvi...	907	907	100%	0.0	99.40%	199234	AY313847.1
Vaccinia virus strain Dryvax clone DPP16, complete genome	Orthopoxvi...	907	907	100%	0.0	99.40%	198820	JN654982.1

The BLAST validation confirmed that the curated sequences are specific to their respective viruses. For each representative sequence:

- Variola sequences matched primarily with other Variola sequences, showing 100% identity and full query coverage. Only a few hits aligned with closely related Orthopoxviruses like Taterapox or Camelpox, reflecting natural sequence similarity within the genus.
- Cowpox sequences showed high identity (around 98–100%) with other Cowpox sequences, confirming their specificity.
- Monkeypox sequences matched exclusively with other Monkeypox sequences, showing low redundancy and high diversity within the species.
- Vaccinia sequences aligned strongly with other Vaccinia sequences, showing high identity and full coverage, with minimal alignment to unrelated sequences.

Across all viruses, no significant hits were observed with unrelated organism. These results demonstrate that the nonredundant sequences are highly specific to each virus.