

VU LEARNING HUB

vulearninghub.xyz

HANDOUT

BIF601

Bioinformatics Computing I

BIF601 by VU Learning HUB

Prepared for VU students • Free study resources

<https://vulearninghub.xyz>

BIF601

Bioinformatics Computing I

STUDY HANDOUT

A complete topic-wise study handout for BIF601 Bioinformatics Computing I, prepared from bioinformatics lecture series. Covers the biology-computing interface, biological databases, sequence alignment, genomics, phylogenetics and protein structure prediction - with algorithms, tools, worked logic and exam pointers.

VU Learning HUB | vulearninghub.xyz

Table of Contents

1. What Is Bioinformatics
2. Molecular Biology Refresher
3. Biological Databases I: Sequence Repositories
4. Biological Databases II: Protein and Specialised
5. Database Searching: Entrez and SRS
6. Pairwise Sequence Alignment I: Concepts
7. Pairwise Alignment II: Dot Plots and Scoring
8. Pairwise Alignment III: Dynamic Programming
9. BLAST: Fast Database Searching
10. Multiple Sequence Alignment
11. Genomics: Sequencing Technologies
12. Gene Prediction and Annotation
13. Phylogenetics I: Trees and Evolution
14. Phylogenetics II: Methods
15. Protein Structure I: Levels and Prediction
16. Protein Structure II: Visualisation and Analysis
17. Practical Bioinformatics: A Workflow
18. Revision: Midterm Practice Set
19. Revision: Final-Term Practice Set

Lecture 1: What Is Bioinformatics

Key terms: bioinformatics, computational biology, genomics

Bioinformatics applies computing to biological data - storing, retrieving and analysing DNA, RNA and protein information. It was born when sequencing began producing data faster than humans could read: the Human Genome Project (3 billion bases) made computation mandatory. Computational biology is the wider twin: modelling biological systems mathematically. Together they drive drug discovery, diagnostics and biotechnology.

- Biology generates big data; computing makes it knowledge.
- Central dogma: DNA → RNA → protein - the data flow bioinformatics follows.
- Applications: gene finding, drug targets, disease mutations, evolutionary trees.

EXAM POINTER: Define bioinformatics and explain why the Human Genome Project needed it.

Lecture 2: Molecular Biology Refresher

Key terms: DNA, RNA, protein, central dogma, codons

DNA's four bases (A, T, G, C) pair A-T and G-C in the double helix; RNA uses uracil instead of thymine. Genes are transcribed to mRNA and translated via triplet codons - 64 codons encode 20 amino acids plus stop signals. Proteins are amino-acid chains folding into functional 3D shapes. This alphabet (4 letters, 20 letters) is what algorithms process as strings.

- Complementary pairing is the basis of PCR, sequencing and hybridisation.
- Codon degeneracy: most amino acids have multiple codons.
- Sequence = string; alignment = string matching with biology-aware scoring.

Lecture 3: Biological Databases I: Sequence Repositories

Key terms: GenBank, EMBL, DDBJ, INSDC, FASTA

The INSDC trio - GenBank (USA), EMBL-EBI (Europe), DDBJ (Japan) - exchange data daily and archive every published DNA sequence with annotations. Records carry accession numbers (stable identifiers), the sequence itself, and metadata (organism, authors, features). The FASTA format (>header line, then sequence lines) is the universal currency for sequence tools.

- Accession numbers: cite these, not bare sequences.
- Annotation quality varies - curated entries beat automated ones.
- FASTA: one header line starting with >, then the raw sequence.

Lecture 4: Biological Databases II: Protein and Specialised

Key terms: UniProt, Swiss-Prot, TrEMBL, PDB, Pfam

UniProtKB/Swiss-Prot provides manually curated protein sequences with function annotations; TrEMBL holds automated translations awaiting curation. The PDB (Protein Data Bank) archives 3D structures solved by X-ray crystallography, NMR and cryo-EM. Pfam classifies protein families by domains - recognising a domain predicts a protein's function. Database choice shapes result quality.

- Swiss-Prot: small, curated, trustworthy; TrEMBL: huge, automated, noisy.
- PDB entries have 4-letter codes (e.g. 1TIM).
- Always check the evidence codes behind an annotation.

Lecture 5: Database Searching: Entrez and SRS

Key terms: Entrez, query, Boolean search, accession

Entrez (NCBI) federates searching across GenBank, PubMed, protein and structure databases with linked records - a gene links to its protein, structure and literature. Effective searching uses field tags (organism, gene name), Boolean operators (AND/OR/NOT), and filters (sequence length, molecule type). Precision beats recall: a tight query returns ten useful records, not ten thousand.

- Use [organism] and [gene] field qualifiers to cut noise.
- Save searches and set alerts for growing gene families.
- Cross-database links turn one hit into a full biological picture.

Lecture 6: Pairwise Sequence Alignment I: Concepts

Key terms: alignment, homology, identity, similarity, gaps

Alignment arranges two sequences to reveal their evolutionary relationship: matching positions suggest common ancestry (homology). Identity counts exact matches; similarity counts chemically similar substitutions. Gaps (insertions/deletions) are penalised because indels are rarer than substitutions. A good alignment maximises matches while minimising gaps - biology expressed as optimisation.

- Homology is an evolutionary claim; similarity is just a number.
- Twilight zone: below ~25% identity, homology needs extra evidence.
- Alignment quality decides everything downstream - trees, models, primers.

Lecture 7: Pairwise Alignment II: Dot Plots and Scoring

Key terms: dot plot, substitution matrix, PAM, BLOSUM

Dot plots visualise similarity: diagonals reveal matching regions, breaks reveal indels. Scoring makes it quantitative: substitution matrices score each aligned pair - PAM (point accepted mutations, evolutionary model) and BLOSUM (blocks-based, empirical). BLOSUM62 is the default for proteins: identities score high, rare substitutions score low, absurd ones negative.

- BLOSUM62: use for general protein searches; BLOSUM80 for close relatives.
- DNA alignments often use simple match/mismatch scores (+5/-4).
- Matrix choice changes results - know why you chose one.

EXAM POINTER: Explain substitution matrices and when to use BLOSUM62 vs BLOSUM80.

Lecture 8: Pairwise Alignment III: Dynamic Programming

Key terms: Needleman-Wunsch, Smith-Waterman, dynamic programming

Dynamic programming guarantees the optimal alignment. Needleman-Wunsch (global) aligns full-length sequences - for similar-length homologs. Smith-Waterman (local) finds the best matching subsequence - for domains shared by otherwise different proteins. Both fill a scoring matrix via recurrence, then trace back. Cost: $O(n*m)$ time and memory - fine for pairs, impossible for databases.

- Global: compare complete genes of similar length.
- Local: find a conserved domain inside long sequences.
- Optimal but slow - hence heuristics like BLAST for searching.

Lecture 9: BLAST: Fast Database Searching

Key terms: BLAST, heuristic, E-value, word size

BLAST trades guaranteed optimality for speed: it seeds on short exact word matches, extends them, and evaluates with statistics. The E-value is the key number - expected random hits this good: $E < 1e-5$ is strong evidence, E near 1 is noise. Variants: blastn (DNA), blastp (protein), blastx (translated DNA vs protein), tblastn, PSI-BLAST (iterative, finds distant homologs).

- Read the E-value first, percent identity second.
- Low-complexity regions (repeats) inflate scores - mask them.
- PSI-BLAST builds a profile and iterates - powerful for remote homology.

Lecture 10: Multiple Sequence Alignment

Key terms: MSA, Clustal, progressive alignment, profiles

Multiple alignment aligns many related sequences at once, revealing conserved columns (functional sites) and variable ones. Progressive methods (Clustal Omega, MUSCLE, MAFFT) align the most similar pairs first, then add sequences - fast but errors propagate. Use MSA to find motifs, design degenerate primers, and as input to phylogenetic trees.

- Guide tree order decides progressive alignment quality.
- Conserved columns = likely functional; check them first.
- Trim unreliable regions before tree-building.

Lecture 11: Genomics: Sequencing Technologies

Key terms: Sanger, NGS, Illumina, assembly

Sanger sequencing (chain termination) gave accurate long reads for the first genomes. Next-generation sequencing (Illumina) produces millions of short reads cheaply - enabling resequencing, RNA-seq and metagenomics. Assembly stitches reads into genomes: overlap-layout-consensus or de Bruijn graphs. Annotation then finds genes - the bridge from raw genome to biology.

- Short reads: cheap, massive; assembly of repeats remains hard.
- Coverage depth (30x) measures sequencing redundancy.
- Third-generation (PacBio, Nanopore) gives long reads closing gaps.

EXAM POINTER: Compare Sanger and NGS sequencing and explain genome assembly.

Lecture 12: Gene Prediction and Annotation

Key terms: gene prediction, ORF, annotation, ab initio

Finding genes in raw DNA: open reading frames (ORFs - start to stop codon stretches) flag candidates; ab initio predictors (Genscan) use statistical signals (splice sites, codon bias); homology methods transfer annotation from known genes. Eukaryotic prediction is harder - introns interrupt ORFs. Annotation adds function: BLAST hits, domain matches, GO terms.

- Prokaryote gene finding is easy (dense ORFs); eukaryotes are hard (introns).
- Combine evidence: no single predictor is reliable alone.
- GO (Gene Ontology) terms standardise function descriptions.

Lecture 13: Phylogenetics I: Trees and Evolution

Key terms: phylogeny, tree, clade, ortholog, paralog

Phylogenetic trees depict evolutionary relationships: leaves are sequences, internal nodes ancestors, branch lengths evolutionary distance. Orthologs (same gene, different species -

speciation) vs paralogs (duplicated within a genome) - only orthologs reliably share function. Trees are hypotheses: different data and methods can disagree.

- Rooted trees show direction of evolution; unrooted show relationships only.
- Outgroup rooting: include a distant relative to place the root.
- Gene trees and species trees can differ - know which you built.

Lecture 14: Phylogenetics II: Methods

Key terms: distance methods, maximum parsimony, maximum likelihood

Tree methods: distance-based (Neighbour-Joining - fast, good for large data), parsimony (fewest changes - simple, can mislead with unequal rates), maximum likelihood and Bayesian (statistical models of evolution - most rigorous, slowest). Bootstrap values (resampling support) measure clade reliability: >70% is respectable, >95% strong.

- Neighbour-Joining for quick exploratory trees.
- ML/Bayesian for publishable phylogenies with model testing.
- Long-branch attraction: fast-evolving sequences falsely group - use better models.

Lecture 15: Protein Structure I: Levels and Prediction

Key terms: primary, secondary, tertiary, quaternary, folding

Structure levels: primary (amino-acid sequence), secondary (alpha-helices, beta-sheets), tertiary (3D fold), quaternary (multi-chain assembly). Anfinsen's principle: sequence determines structure. Prediction approaches: homology modelling (template-based - reliable when identity >30%), threading, and ab initio/AI methods (AlphaFold - revolutionary accuracy from sequence alone).

- Structure is more conserved than sequence - fold reveals distant homology.
- Homology modelling needs a good template alignment.
- AlphaFold changed the field: validate predictions experimentally still.

Lecture 16: Protein Structure II: Visualisation and Analysis

Key terms: PDB, PyMOL, Ramachandran, domains

PDB files store atomic coordinates; viewers (PyMOL, Chimera) render cartoons, surfaces and sticks. The Ramachandran plot validates backbone geometry - outliers flag errors. Domain analysis (Pfam, SMART) decomposes proteins into functional modules. Structure comparison (DALI) finds similar folds regardless of sequence - function follows form.

- Cartoon view: helices as coils, sheets as arrows.
- Active sites: conserved 3D clusters, often predictable from structure.

- Download a PDB file and identify its domains - standard practical.

EXAM POINTER: Describe the four levels of protein structure and how AlphaFold predicts them.

Lecture 17: Practical Bioinformatics: A Workflow

Key terms: workflow, pipeline, reproducible research

A typical project: obtain sequences (database), quality-check, align (MSA), build a tree, annotate function, visualise. Tools chain via scripts (Python/Biopython, R) for reproducibility - point-and-click doesn't scale or document. Version your data and code; record parameters (BLAST E-values, matrices) so results can be reproduced and defended.

- Biopython: parse FASTA, run BLAST, handle alignments in Python.
- Script everything - manual steps are irreproducible steps.
- Report parameters with results; defaults are choices too.

Lecture 18: Revision: Midterm Practice Set

1. Describe the INSDC databases and the FASTA format. (10 marks)
2. Explain substitution matrices: PAM vs BLOSUM. (10 marks)
3. Compare Needleman-Wunsch and Smith-Waterman algorithms. (10 marks)
4. A BLAST hit shows $E=1e-40$, 85% identity. Interpret fully. (5 marks)
5. What is multiple sequence alignment and what is it used for? (10 marks)

Lecture 19: Revision: Final-Term Practice Set

1. Compare Sanger and NGS sequencing; outline genome assembly. (10 marks)
2. Explain orthologs vs paralogs and why the distinction matters. (5 marks)
3. Compare NJ, parsimony and maximum likelihood tree methods. (10 marks)
4. Describe homology modelling and the role of AlphaFold. (10 marks)
5. Outline a reproducible bioinformatics workflow for a new gene. (10 marks)

End of Handout - Good luck with your exams!