

VU LEARNING HUB

vulearninghub.xyz

HIGHLIGHTED HANDOUT

MTH645

Fuzzy Logic and Applications

MTH645 by VU Learning HUB

Prepared for VU students • Free study resources

<https://vulearninghub.xyz>

MTH645

Fuzzy Logic and Applications

Lecture-Wise Study Handout
Virtual University of Pakistan

22 lectures • prepared for exam study 

Table of Contents

1. Crisp vs Fuzzy Sets
2. Membership Functions and Linguistic Variables
3. Fuzzy Set Operations
4. Fuzzy Relations and Composition
5. Fuzzy Numbers and Arithmetic
6. Fuzzy Inference Systems
7. Defuzzification and Fuzzy Control
8. Clustering, Classification, and Decisions
9. Hybrid Systems and Synthesis
10. Fuzziness vs Probability; Approximate Reasoning
11. Worked Example: Fuzzy Washing Machine
12. Fuzzy Databases and Optimization
13. Fuzzy Measures and Multicriteria Decisions
14. Applications Across Disciplines
15. Exam Drills and Revision 
16. Solved: Operations and α -Cuts
17. Solved: Relations and Inference Traces
18. Case Studies: Fuzzy vs Crisp
19. Question Bank and Final Revision
20. Fuzzy Clustering
21. Neuro-Fuzzy Systems (ANFIS)
22. Fuzzy Time Series Forecasting

About This Course

MTH645 – Fuzzy Logic and Applications studies reasoning with vagueness: from Lotfi Zadeh's 1965 fuzzy sets through membership functions, fuzzy operations, inference systems, and defuzzification, to real control and decision applications. This handout balances the mathematics (definitions, operations, proofs of key properties) with worked examples and applications.

Each lecture gives learning objectives, core explanations with definitions, worked examples, exam key  points, a summary, and review questions.

Lecture 1: Crisp vs Fuzzy Sets

Learning Objectives

- Define fuzzy sets
- Describe membership structure

Classical (crisp) sets have binary membership: an element is in or out — the characteristic function's values are $\{0,1\}$. This bivalence fails for vague concepts: is 35°C 'hot'? Is a 5'11" man 'tall'? Forcing yes/no loses information. Lotfi Zadeh's 1965 insight: let membership be a matter of degree — a fuzzy set maps each element to a membership grade in $[0,1]$.

A fuzzy set A on universe X is defined by its membership function $\mu_A: X \rightarrow [0,1]$. Example: 'tall men' with $\mu(x) = 0$ for $x \leq 5'6"$, rising linearly to 1 at 6'2". A man of 5'11" might have $\mu = 0.7$ — mostly tall. The support ($\mu > 0$), core ($\mu = 1$), and crossover points ($\mu = 0.5$) describe the set's shape; height is its maximum membership.

Why it matters: much human reasoning is graded — controllers, expert systems, and decision models built on fuzzy sets handle vagueness natively instead of pretending precision. The exam-ready line:  fuzzy logic is not imprecise logic — it is precise reasoning about imprecise concepts.

Key Definitions

Fuzzy set is defined as a set with graded membership: $\mu_A(x) \in [0,1]$ for each element x .

Examples

- 'Hot' temperature: $\mu = 0$ at 20°C, 0.5 at 30°C, 1 at 40°C.
- Crisp failure: a 5'10.9" man 'not tall' but 5'11" 'tall' — the boundary absurdity.

Key Points — very important for exams

Crisp: $\{0,1\}$ membership; fuzzy: $[0,1]$ grades — Zadeh 1965.

Membership function μ_A defines the set; support/core/crossover describe it.

Vague concepts (tall, hot) need grades, not boundaries.

Fuzzy logic = precise reasoning about imprecise concepts.

Note:

Note: 'fuzzy logic is not imprecise logic' is the quotable exam line — use it to open or close answers.

Review Questions

- ? Is a fuzzy set with all grades 0/1 just a crisp set?
- ? Sketch μ for 'middle-aged'.

Section Summary

Fuzzy sets replace membership boundaries with membership grades — the foundation of everything in this course.

Exam Pointers

- Distinguish crisp and fuzzy sets.
- Define support, core, and crossover points.

Lecture 2: Membership Functions and Linguistic Variables

Learning Objectives

- Choose membership functions
- Use linguistic variables and hedges

Standard membership functions: triangular (a,b,c) — simple, the workhorse; trapezoidal (a,b,c,d) — flat tops for 'about the same' regions; Gaussian — smooth, for natural variation; sigmoidal/S-curves — for 'large/small' extremes; singleton — a spike, for crisp inputs. Choice is part art: triangular/trapezoidal for interpretability, Gaussian for smoothness.

Linguistic variables: variables whose values are words — temperature $\in$ {cold, warm, hot}, each a fuzzy set. This is fuzzy logic's human interface: rules written in words ('if temperature is hot then fan speed is high'), computed with mathematics. Hedges modify meaning: very A = μA^2 (concentration), more-or-less A = $\sqrt{\mu A}$ (dilation), not A = $1 - \mu A$.

Example: error $e \in$ {Negative, Zero, Positive} as triangular sets on $[-10,10]$; 'very Positive' squares the Positive curve, shrinking partial memberships. Exam skill: given a universe and linguistic terms, sketch the membership functions and compute hedged values at a point.

Key Definitions

Linguistic variable is defined as a variable taking linguistic values (words), each modeled as a fuzzy set.

Hedges is defined as modifiers like 'very' (concentration, μ^2) that reshape membership functions.

Examples

- 'Very hot' = μ_{hot}^2 : $0.7 \rightarrow 0.49$ — concentration.
- Linguistic variable: pressure $\in$ {low, medium, high}.

Key Points — very important for exams

Triangular/trapezoidal/Gaussian/sigmoidal/singleton — know each shape.

Linguistic variables: words as fuzzy sets — the human interface.

Hedges: very = square, more-or-less = square root, not = complement.

Sketch-and-compute at a point: the core exam skill.

Review Questions

- ? Sketch trapezoidal 'comfortable' temperature.
- ? Compute 'very warm' at $\mu_{\text{warm}} = 0.64$.

Section Summary

Membership functions mathematize words; hedges adjust them — together they let rules speak human and compute precisely.

Exam Pointers

- What are the standard membership functions?
- How do hedges work mathematically?

Lecture 3: Fuzzy Set Operations

Learning Objectives

- Apply fuzzy operations
- Test classical laws

Fuzzy set operations extend crisp ones: complement $\mu_{\neg A} = 1 - \mu_A$; intersection (t-norms): standard $\min(\mu_A, \mu_B)$, algebraic product $\mu_A \cdot \mu_B$; union (t-conorms): standard $\max(\mu_A, \mu_B)$, algebraic sum $\mu_A + \mu_B - \mu_A \cdot \mu_B$. The standard min/max are the most used — idempotent ($A \cap A = A$) and intuitively 'and/or'-like.

Properties: commutativity, associativity, distributivity hold for min/max; but the excluded-middle laws fail — $A \cup \neg A \neq X$ and $A \cap \neg A \neq \emptyset$ in general. This is not a bug: a 0.5-membered element is half in A and half in its complement — exactly what vagueness means. De Morgan's laws survive: complement of union = intersection of complements.

Worked example: A = tall (μ : Ali 0.7), B = young (μ : Ali 0.4). Standard: tall AND young = min = 0.4; tall OR young = max = 0.7; NOT tall = 0.3. Algebraic product: 0.28; algebraic sum: 0.82. Exam skill: compute both standard and algebraic operations at given points, and state which laws hold or fail.

Key Definitions

t-norms is defined as binary operations generalizing AND (min, product) to fuzzy sets.

Examples

- $\min(0.7, 0.4) = 0.4$ vs product 0.28: standard is 'softer'.
- Excluded middle: $\mu = 0.5$ gives $A \cup \neg A$ grade 0.5, not 1.

Key Points — very important for exams

Complement $1 - \mu$; AND via t-norms (min, product); OR via t-conorms (max, sum).

Min/max: idempotent, intuitive — the default choice.

Excluded middle fails — vagueness, not a bug; De Morgan survives.

Compute at points; know which laws hold.

Review Questions

? Prove De Morgan for fuzzy sets with min/max.

? When is algebraic product preferable to min?

Section Summary

Fuzzy operations generalize AND/OR/NOT to grades: min/max as standard, with the excluded middle gracefully surrendered.

Exam Pointers

→ Compute fuzzy AND/OR with min/max and algebraic operators.

→ Which classical laws fail for fuzzy sets, and why?

Lecture 4: Fuzzy Relations and Composition

Learning Objectives

- Define fuzzy relations
- Compose them

A fuzzy relation R on $X \times Y$ assigns each pair (x, y) a grade $\mu_R(x, y) \in [0, 1]$ — e.g., 'x is much larger than y' with grades rising in the difference. Relations encode rules and associations: the rule 'if temperature is hot then fan is fast' is a relation between temperature and fan-speed universes.

Composition: $(R \circ S)(x, z) = \max_y \min(\mu_R(x, y), \mu_S(y, z))$ — max-min composition, the workhorse of fuzzy inference. Properties: max-min composition is associative and distributes over union; the inverse relation swaps arguments. Powers of relations ($R^2 = R \circ R$) model indirect association.

Worked example: R (x is close to y) on $\{1, 2, 3\}$, S (y is close to z): compose to find how close x is to z through y. Fuzzy graphs and similarity relations (reflexive, symmetric, max-min transitive) cluster data — the basis of fuzzy clustering (next lectures). Exam skill: compute small max-min compositions by hand.

Key Definitions

Max-min composition is defined as $(R \circ S)(x, z) = \max_y \min(\mu_R(x, y), \mu_S(y, z))$ — chaining fuzzy relations.

Examples

■ Rule as relation: 'hot $\rightarrow$ fast' as a fuzzy Cartesian product.

■ Transitivity: similarity relations cluster via max-min powers.

Key Points — very important for exams

Relations grade pairs (x,y) — rules and associations as relations.

Max-min composition chains relations: $\max_y \min(\dots)$.

Associative, distributes over union; inverse swaps arguments.

Similarity relations (reflexive/symmetric/transitive) enable clustering.

Review Questions

- ? Compose two 2×2 fuzzy relations by hand.
- ? Why is max-min natural for chaining?

Section Summary

Relations turn rules into mathematics; max-min composition turns rule-chains into inference — the engine room of fuzzy systems.

Exam Pointers

- What is a fuzzy relation?
- Compute a max-min composition.

Lecture 5: Fuzzy Numbers and Arithmetic

Learning Objectives

- Define fuzzy numbers
- Compute with α -cuts

Fuzzy numbers generalize real numbers: a fuzzy number is a convex, normalized fuzzy set on $\mathbb{R}$ — 'about 5' as a triangular $(4,5,6)$. Convexity (membership rises then falls, no dips) keeps arithmetic sane. α -cuts (the crisp sets $\{x : \mu(x) \geq \alpha\}$) are the computational trick: do interval arithmetic per α -cut, then reassemble — the extension principle made practical.

Fuzzy arithmetic via α -cuts: $[a_1, a_2] + [b_1, b_2] = [a_1 + b_1, a_2 + b_2]$; multiplication and division similarly on positive intervals. Example: 'about 3' $(2,3,4) +$ 'about 5' $(4,5,6) =$ 'about 8' $(6,8,10)$ — triangular preserved under addition (multiplication only approximately).

Applications: engineering tolerances ('the beam is about 5m'), project durations (fuzzy PERT — 'about 3 weeks'), and cost estimation under vagueness. Exam skill: add/subtract triangular fuzzy numbers via α -cuts and sketch the result.

Key Definitions

α -cut is defined as the crisp set of elements with membership $\geq \alpha$ — the discretization tool for fuzzy arithmetic.

Examples

- 'About 5' = (4,5,6); its 0.5-cut is [4.5,5.5].
- Fuzzy PERT: activity 'about 3 weeks' in project scheduling.

Key Points — very important for exams

Fuzzy numbers: convex, normalized sets on $\mathbb{R}$ — 'about n'.

α -cuts convert fuzzy arithmetic to interval arithmetic.

Addition preserves triangular shape; multiplication approximately.

Use: tolerances, fuzzy PERT, vague cost estimation.

Review Questions

- ? Find the 0.25-cut of triangular (2,4,6).
- ? Why require convexity?

Section Summary

α -cuts make fuzzy arithmetic computable: slice at each grade, do interval math, reassemble the fuzzy answer.

Exam Pointers

- What is a fuzzy number?
- Add two triangular fuzzy numbers via α -cuts.

Lecture 6: Fuzzy Inference Systems

Learning Objectives

- Build a FIS
- Trace inference

A fuzzy inference system (FIS) turns inputs into outputs via rules: fuzzification (crisp input $\rightarrow$ membership grades), rule evaluation (each rule's firing strength — min of antecedent grades), aggregation (combining rule outputs — max), and defuzzification (fuzzy output $\rightarrow$ crisp number). The classic example: tip = f(service, food) — 'if service is poor or food is rancid then tip is cheap'.

Mamdani inference: rule consequents are fuzzy sets; outputs aggregate by max and defuzzify (centroid). Intuitive and interpretable — the standard for control and expert systems. Worked trace: service=3/10 (poor: 0.7), food=8/10 (delicious: 0.8); rule 'poor service $\rightarrow$ cheap tip' fires at 0.7; the cheap-tip set clipped at 0.7; centroid of aggregated output $\approx$ the tip.

Sugeno inference: consequents are functions (typically linear: $z = px + qy + r$); output is the weighted average of rule outputs — no defuzzification step, computationally efficient, ideal for adaptive/neuro-fuzzy systems. Exam comparison: Mamdani = interpretable, Sugeno = efficient —

choose by need.

Key Definitions

Fuzzification is defined as converting a crisp measurement into membership grades in input fuzzy sets.

Defuzzification is defined as converting the aggregated fuzzy output into a single crisp number (e.g., centroid).

Examples

■ Tipper: service 3 → poor 0.7; rule fires 0.7; centroid $\approx$ 7% tip.

■ Sugeno: $z = 2x + y + 1$ per rule; weighted average output.

Key Points — very important for exams

Pipeline: fuzzify → evaluate rules → aggregate → defuzzify.

Mamdani: fuzzy consequents, centroid defuzzification — interpretable.

Sugeno: function consequents, weighted average — efficient.

Hand-trace small systems: the core exam skill.

Review Questions

? Defuzzify a triangular output clipped at 0.6 by centroid.

? When is Sugeno preferable?

Section Summary

Fuzzy inference computes with words: grade the inputs, fire the rules, blend the outputs — Mamdani for insight, Sugeno for speed.

Exam Pointers

→ Trace a Mamdani inference by hand.

→ Compare Mamdani and Sugeno.

Lecture 7: Defuzzification and Fuzzy Control

Learning Objectives

- Defuzzify
- Design fuzzy controllers

Defuzzification methods: centroid (center of gravity — the standard, smooth but computation-heavy), bisector (splits area in half), mean of maximum (average of points with max membership — fast, jumpy), smallest/largest of maximum. Choice affects behavior: centroid for smooth control, MOM for

fast decisions.

Fuzzy control architecture: error and change-of-error as inputs ('error is Positive Small'), control action as output ('increase heating Slightly'), with a rule base covering the input space (typically 25–49 rules for two inputs $\times$ 5–7 terms). Design steps: define universes and linguistic terms, write the rule table (the expert knowledge), choose inference/defuzzification, then tune (adjust membership functions and rules against performance).

Classic applications: Sendai subway (1987 — smoother braking than human drivers), washing machines (dirtiness/load $\rightarrow$ wash time), air conditioners, and autofocus cameras. Exam skill: given a  3 $\times$ 3 rule table and inputs, compute the crisp control action.

Key Definitions

Centroid is defined as the center-of-gravity of the output fuzzy set — the standard defuzzification method.

Examples

- AC controller: error $\{N, Z, P\} \times \Delta\text{error} \{N, Z, P\} \rightarrow 9 \text{ rules} \rightarrow \text{fan speed}$.
- Sendai: fuzzy braking smoother than PID — the famous success.

Key Points — very important for exams

Centroid (smooth), MOM (fast), bisector — match to need.

Controller: error $\times$ $\Delta\text{error} \rightarrow$ rule table $\rightarrow$ crisp action.

Design: universes $\rightarrow$ terms $\rightarrow$ rules $\rightarrow$ tune against performance.

Sendai subway, washing machines: the classic wins.

Review Questions

- ? Compute centroid of a simple aggregated output.
- ? Write 9 rules for a water-level controller.

Section Summary

Fuzzy control bottles expertise into rule tables: grade the error, consult the rules, act crisply — smoothly.

Exam Pointers

- $\rightarrow$ Compare defuzzification methods.
- $\rightarrow$ Design a fuzzy controller for room temperature.

Lecture 8: Clustering, Classification, and Decisions

Learning Objectives

- Cluster fuzzily
- Decide with fuzzy goals

Fuzzy clustering: data points belong to clusters by degree — the Fuzzy C-Means (FCM) algorithm minimizes the weighted within-cluster variance, iterating: compute memberships from distances, recompute centroids from memberships, until convergence. Unlike k-means (hard assignment), FCM handles overlapping clusters — a customer can be 0.7 'budget' and 0.3 'premium'.

Pattern recognition and classification: fuzzy classifiers output class membership grades — useful in medical diagnosis (symptoms suggest diseases by degree) and image segmentation (pixels partially in regions — the fuzzy approach to noisy images). Fuzzy decision-making: Bellman-Zadeh — decisions maximize the intersection of goals and constraints as fuzzy sets ('find the option maximizing $\min(\text{goal}, \text{constraint})$ satisfaction').

Pakistani-relevant applications: crop-yield prediction under vague weather data, load-shedding scheduling with fuzzy demand, and Urdu text classification. Exam point: wherever boundaries are naturally graded, fuzzy methods outperform crisp forcing. 

Key Definitions

Fuzzy C-Means is defined as clustering where points hold membership grades across clusters — Bezdek's Fuzzy C-Means.

Examples

- FCM: a borderline customer 0.6/0.4 across segments — not forced into one.
- Medical: symptom set suggests dengue 0.8, flu 0.3 — graded diagnosis.

Key Points — very important for exams

FCM: graded cluster membership via iterative optimization.

Fuzzy classifiers: graded diagnosis, image segmentation.

Bellman-Zadeh: decide by maximizing $\text{goal} \cap \text{constraint}$ satisfaction.

Use where boundaries are naturally graded.

Review Questions

- ? One FCM iteration by hand on 4 points.
- ? When does fuzzy beat crisp classification?

Section Summary

From clusters to diagnoses to decisions: fuzzy methods grade the answers where crisp methods force them.

Exam Pointers

- How does Fuzzy C-Means differ from k-means?

→ What is Bellman-Zadeh decision-making?

Lecture 9: Hybrid Systems and Synthesis

Learning Objectives

- Explain hybrid systems
- Synthesize the course

Neuro-fuzzy systems (e.g., ANFIS — Adaptive Neuro-Fuzzy Inference System): a Sugeno FIS structured as a neural network, its membership-function parameters learned from data by gradient descent + least squares. The payoff: interpretability of fuzzy rules with the learning power of neural nets — 'gray box' modeling.

Genetic fuzzy systems: evolutionary algorithms tune membership functions and rule bases — useful when gradients are unavailable. Fuzzy expert systems: MYCIN-style medical diagnosis, fault detection in plants — rules elicited from experts, executed fuzzily. Type-2 fuzzy sets (membership grades themselves fuzzy) handle uncertainty about the membership functions — for very noisy domains.

Course synthesis: fuzzy logic's arc — from Zadeh's sets (1965) through control triumphs (1980s–90s) to hybrid intelligent systems today. Exam-ready close: fuzzy logic succeeded not by replacing probability or crisp logic but by formalizing the vague reasoning humans actually use.

Key Definitions

ANFIS is defined as a Sugeno FIS as a neural network, learning membership parameters from data.

Examples

- ANFIS: learning 'hot' and 'fast' curves from operating data.
- Type-2: sensor noise making even the membership grades uncertain.

Key Points — very important for exams

ANFIS: fuzzy rules + neural learning — interpretable and adaptive.

Genetic tuning, fuzzy expert systems, type-2 sets for meta-uncertainty.

Arc: 1965 sets → 1980s control → today's hybrids.

Fuzzy logic formalizes vague human reasoning — its enduring contribution.

Note:

Note: this synthesis — 'not replacing probability but formalizing vagueness' — is the ideal closing line for essay answers.

Review Questions

? When would you choose ANFIS over a neural net?

? Is fuzzy logic a rival to probability? Discuss.

Section Summary

From Zadeh's 1965 paper to learning hybrids: fuzzy logic endures because human reasoning is graded — and now it's mathematics.

Exam Pointers

- What is a neuro-fuzzy system?
- How do type-2 fuzzy sets differ?

Lecture 10: Fuzziness vs Probability; Approximate Reasoning

Learning Objectives

- Separate the two uncertainties
- Apply approximate reasoning

Fuzzy logic vs probability — the perennial confusion: probability quantifies uncertainty about crisp events (will it rain? — yes/no, unknown); fuzziness quantifies vagueness in the event itself (is it 'hot'? — a matter of degree, fully observed). A 0.7 membership in 'hot' is not a 70% chance of hot — the temperature is known exactly; its hotness is graded.

They complement: use probability for randomness, fuzziness for vagueness — and fuzzy-probabilistic hybrids exist (fuzzy random variables). Exam trap: 'the probability of a fuzzy event' (Zadeh defined it: the expected membership) vs 'fuzzy probability' — related but distinct concepts.

Approximate reasoning: generalized modus ponens — from 'if x is A then y is B' and 'x is A' (A' close to A), conclude 'y is B' (B' close to B). This is how fuzzy controllers 'reason': partial matches fire rules partially. Compositional rule of inference formalizes it via max-min composition of the fact with the rule-relation.

Key Definitions

Generalized modus ponens is defined as Zadeh's inference pattern: partial matching of premises yields partially-matched conclusions.

Examples

- Known 35°C day: hotness 0.8 (fuzzy), rain chance 30% (probability) — different uncertainties.
- Generalized MP: 'temperature somewhat hot' → 'fan somewhat fast'.

Key Points — very important for exams

Probability = uncertainty about crisp events; fuzziness = graded events.

0.7 membership $\neq$ 70% probability — never confuse them.

Generalized modus ponens: partial matches, partial conclusions.

Compositional rule: max-min of fact with rule-relation.

Review Questions

? A doctor says 'likely' vs 'somewhat severe' — which uncertainty is which?

? Trace one generalized-MP step.

Section Summary

Fuzziness and probability answer different questions; approximate reasoning lets partial truths propagate soundly.

Exam Pointers

→ Distinguish fuzziness from probability.

→ What is the compositional rule of inference?

Lecture 11: Worked Example: Fuzzy Washing Machine

Learning Objectives

- Trace a complete FIS
- Extract design lessons

Complete worked example — washing machine: inputs dirtiness $d \in [0,10]$ (sensors) and load $l \in [0,10]$; output wash time $t \in [0,30]$ min. Linguistic terms: dirtiness {Clean, Medium, Dirty}, load {Light, Normal, Heavy}, time {Short, Medium, Long} — triangular sets. Rule table (9 rules): e.g., Dirty $\wedge$ Heavy $\rightarrow$ Long; Clean $\wedge$ Light $\rightarrow$ Short.

Trace with $d=7, l=8$: $\mu_{\text{Dirty}}(7)=0.8, \mu_{\text{Heavy}}(8)=0.75 \rightarrow$ rule fires at $\min=0.75 \rightarrow$ Long clipped at 0.75. Suppose also Medium-dirtiness fires 0.2 $\rightarrow$ Medium clipped 0.2. Aggregate (max), centroid ≈ 22 min. The machine washes longer for dirtier, heavier loads — smoothly, with no thresholds.

Design lessons: rule tables encode expert heuristics; smoothness comes from overlapping sets; tuning = adjusting sets/rules against test loads. Exam skill: full hand-trace with given sets and inputs — show fuzzification, firing, aggregation, centroid.

Key Definitions

Rule table is defined as the expert-knowledge table mapping input-term combinations to output terms.

Examples

■ $d=7, l=8 \rightarrow \sim 22$ min: the complete pipeline in numbers.

■ Threshold failure: crisp 'if dirt > 7 then 25 min' jerks at 6.9 vs 7.1.

Key Points — very important for exams

Inputs: dirtiness $\times$ load $\rightarrow$ 9-rule table $\rightarrow$ wash time.

Hand-trace: fuzzify $\rightarrow$ min-firing $\rightarrow$ clip $\rightarrow$ max-aggregate $\rightarrow$ centroid.

Overlapping sets give smooth control; crisp thresholds jerk.

Tune sets and rules against performance data.

Note:

Note: a full hand-trace like this is the single most likely long numerical question — practice it until fluent.

Review Questions

- ? Recompute with $d=4$, $l=3$.
- ? What breaks if sets don't overlap?

Section Summary

The washing machine is fuzzy control in miniature: expert rules, smooth responses, tunable behavior.

Exam Pointers

- $\rightarrow$ Trace a fuzzy washing-machine decision by hand.
- $\rightarrow$ Why are overlapping sets essential?

Lecture 12: Fuzzy Databases and Optimization

Learning Objectives

- Query fuzzily
- Formulate fuzzy LPs

Fuzzy databases: queries with vague predicates — 'find young, well-paid employees' — evaluated by membership grades, ranked by satisfaction degree. Fuzzy SQL extensions compute μ per row and return graded results instead of empty crisp answers ('no employee is exactly young AND well-paid' vs 'Ali satisfies 0.8').

Fuzzy linear programming: coefficients and constraints as fuzzy numbers — 'profit is about Rs. 50/unit', 'capacity is at most about 1000'. Zimmermann's approach: maximize the minimum satisfaction of objectives and constraints (the Bellman-Zadeh max-min again) — solved via crisp LP on α -cuts or auxiliary variables.

Applications: production planning with vague demand, portfolio selection with fuzzy returns, and diet/blending problems with approximate requirements. Exam point: fuzziness enters OR models where data is linguistically specified — the max-min satisfaction pattern recurs.

Key Definitions

Fuzzy queries is defined as database queries with vague predicates evaluated by membership grades and ranked by satisfaction.

Examples

- Query: 'tall, experienced managers' → ranked by $\min(\text{tall}, \text{experienced})$.
- Fuzzy LP: 'about 1000 units capacity' as a trapezoidal constraint.

Key Points — very important for exams

Fuzzy queries: graded predicates, ranked results — no empty answers.

Fuzzy LP: vague coefficients/constraints; maximize min-satisfaction.

Solved via α -cuts or auxiliary-variable crisp LPs.

Pattern: Bellman-Zadeh max-min satisfaction recurs.

Review Questions

- ? Rank three employees on 'young and well-paid'.
- ? Formulate a fuzzy knapsack constraint.

Section Summary

Databases and optimizers go fuzzy where specifications are linguistic: grade satisfaction, maximize the minimum.

Exam Pointers

- How do fuzzy database queries work?
- What is Zimmermann's fuzzy LP approach?

Lecture 13: Fuzzy Measures and Multicriteria Decisions

Learning Objectives

- Apply fuzzy MCDM
- Explain Choquet intuition

Fuzzy measures generalize probability measures (dropping additivity): $g(\emptyset)=0$, $g(X)=1$, monotonicity — modeling importance or interaction among criteria. The Choquet integral aggregates with respect to a fuzzy measure — capturing redundancy (two similar criteria count once) and synergy (complementary criteria count extra) that weighted averages miss.

Multicriteria decision example: hiring on {skill, experience, fit} — skill and experience redundant (overlap), skill and fit synergistic. A fuzzy measure encodes this; Choquet aggregation scores candidates accordingly. This beats naive weighting whenever criteria interact — which is usually.

Simpler tools for exams: fuzzy AHP (pairwise comparisons as fuzzy numbers for vague judgments) and fuzzy TOPSIS (ranking alternatives by distance to fuzzy ideal solutions). The takeaway: fuzzy MCDM handles the vagueness of human judgment in structured decisions.

Key Definitions

Choquet integral is defined as aggregation with respect to a fuzzy measure — capturing criteria interaction (redundancy, synergy).

Examples

- Hiring: skill $\cap$ experience redundant; skill $\cap$ fit synergistic — encoded in g.
- Fuzzy AHP: 'skill is moderately more important than experience' as (2,3,4).

Key Points — very important for exams

Fuzzy measures: monotonic, non-additive — model criteria interaction.

Choquet integral: redundancy and synergy aware aggregation.

Fuzzy AHP/TOPSIS: vague judgments in structured decisions.

Use when criteria interact — which is nearly always.

Review Questions

- ? Why is additivity wrong for criteria?
- ? Rank two candidates with a given fuzzy measure.

Section Summary

Decisions involve interacting, vaguely-judged criteria: fuzzy measures and Choquet integrals aggregate them honestly.

Exam Pointers

- Why do fuzzy measures generalize probability?
- When does Choquet beat weighted averaging?

Lecture 14: Applications Across Disciplines

Learning Objectives

- Survey applications
- Match method to domain

Economics and management applications: fuzzy cost-benefit ('benefits are substantial'), project selection under vague estimates, consumer-behavior models with linguistic preferences ('brand loyalty is high'), and quality control (fuzzy control charts for 'acceptable' quality grades rather than binary pass/fail).

Engineering beyond control: fuzzy image processing (noisy segmentation), structural safety ('the bridge is sufficiently safe' — fuzzy reliability), and traffic signal control (fuzzy logic famously beats fixed timers in variable traffic). Medicine: diagnostic expert systems, anesthesia control, and treatment dosing under vague symptoms.

Social science frontiers: fuzzy poverty measures ('somewhat poor' — better targeting than poverty lines), opinion aggregation, and conflict analysis. Exam thesis: fuzzy methods shine wherever experts think in words and thresholds would lie — which covers most applied domains. 

Key Definitions

Fuzzy poverty measures is defined as poverty measurement by membership grades rather than a binary poverty line.

Examples

- Traffic: fuzzy signals adapting to 'heavy' vs 'light' flow — smoother than timers.
- Anesthesia: dosing from 'patient somewhat responsive' — graded control.

Key Points — very important for exams

Management: vague estimates, linguistic preferences, fuzzy QC.

Engineering: images, structures, traffic, medicine.

Social: fuzzy poverty, opinion, conflict analysis.

Thesis: fuzzy wins where experts think in words and thresholds lie.

Review Questions

- ? Design a fuzzy traffic controller's inputs and rules.
- ? Where would fuzzy methods fail?

Section Summary

From washing machines to poverty policy: wherever vagueness is real, fuzzy formalization beats crisp pretense.

Exam Pointers

- Give five fuzzy applications outside control.
- Why do fuzzy poverty measures beat poverty lines?

Lecture 15: Exam Drills and Revision

Learning Objectives

- Drill computations
- Organize revision

Exam problem drills — Problem 1: Given $A=\{(x1,0.2),(x2,0.7),(x3,0.5)\}$, $B=\{(x1,0.6),(x2,0.3),(x3,0.9)\}$: compute $A \cup B$ (max), $A \cap B$ (min), $\blacksquare$, and $A \cap \blacksquare$ — then verify excluded-middle failure at $x2$. Problem 2: triangular $A=(1,2,3)$, $B=(2,3,4)$: find the 0.5-cuts and add them.

Problem 3: relation R on $\{a,b\} \times \{1,2\} = [[0.8,0.3],[0.2,0.9]]$, S on $\{1,2\} \times \{x,y\} = [[0.7,0.4],[0.5,1.0]]$: compute $R \blacksquare S$ (max-min) — answer $[[0.7,0.4],[0.5,0.9]]$. Problem 4: trace the tipper FIS for service=4, food=6 with given sets — practice the full pipeline.

Revision checklist: definitions (fuzzy set, α -cut, linguistic variable, t-norm, relation, inference, defuzzification) — one line each; key computations (operations at points, α -cut arithmetic, max-min composition, Mamdani trace, centroid) — practice each twice; proofs (De Morgan, associativity) — know the outlines; comparisons (fuzzy vs probability, Mamdani vs Sugeno, FCM vs k-means) — table form.

Key Definitions

Revision checklist is defined as the exam-ready toolkit: definitions, five computation types, proof outlines, comparison tables.

Examples

■ α -cut addition: $A_{0.5}=[1.5,2.5]$, $B_{0.5}=[2.5,3.5] \rightarrow \text{sum } [4,6]$.

■ Tipper trace: the full pipeline as a 10-mark answer.

Key Points — very important for exams

Five computation types: operations, α -cuts, composition, inference trace, centroid.

Proofs: De Morgan, associativity — know outlines.

Comparisons as tables: fuzzy/probability, Mamdani/Sugeno, FCM/k-means.

Practice each computation twice — fluency is marks.

Note:

Note: work every problem here by hand before the exam — fuzzy courses reward computational fluency above all.

Review Questions

? Redo all four problems closed-book.

? Which computation is your weakest? Drill it.

Section Summary

Fuzzy exams test fluent computation: definitions crisp, operations fast, traces complete — drill until automatic.

Exam Pointers

→ Solve: $A \cap \blacksquare$ for the given sets — what law fails?

→ Compute $R \blacksquare S$ for the given matrices.

Lecture 16: Solved: Operations and α -Cuts

Learning Objectives

- Compute fluently
- Apply α -cuts

Problem set 1 — operations: $A = \{(1,0.3), (2,0.8), (3,0.5)\}$, $B = \{(1,0.6), (2,0.4), (3,0.9)\}$. Standard: $A \cap B = \{(1,0.3), (2,0.4), (3,0.5)\}$; $A \cup B = \{(1,0.6), (2,0.8), (3,0.9)\}$; $\blacksquare = \{(1,0.7), (2,0.2), (3,0.5)\}$. Algebraic: $A \cap B$ (product) = $\{(1,0.18), (2,0.32), (3,0.45)\}$; $A \cup B$ (sum) = $\{(1,0.72), (2,0.88), (3,0.95)\}$. Excluded middle: $A \cup \blacksquare$ at $x=2$: $\max(0.8, 0.2) = 0.8 \neq 1$ — fails, as expected.

Problem set 2 — α -cuts: $A = (2, 5, 8)$ triangular: $A_{0.25} = [2 + 0.25(3), 8 - 0.25(3)] = [2.75, 7.25]$; $A_{-1} = \{5\}$. $B = (1, 3, 5)$: $A+B$ via cuts: $[3.75, 12.25]$ at 0.25 — reassembled $\approx (3, 8, 13)$. Scalar: $2A = (4, 10, 16)$. Problem set 3 — hedges: $\mu_{\text{tall}}(\text{Ali}) = 0.64$: 'very tall' = 0.41, 'more-or-less tall' = 0.8, 'not tall' = 0.36.

Exam technique: write the formula first, substitute, compute — markers award method marks. For α -cuts of triangular (a, b, c) : $[a + \alpha(b-a), c - \alpha(c-a)]$ — memorize this linear interpolation. 

Key Definitions

Triangular α -cut formula is defined as the α -cut of triangular (a, b, c) : $[a + \alpha(b-a), c - \alpha(c-a)]$ — linear interpolation.

Examples

- $\blacksquare \cup \blacksquare \neq X$ at 0.8: the excluded-middle failure, demonstrated.
- $\blacksquare (2, 5, 8) + (1, 3, 5) = (3, 8, 13)$: triangular addition.

Key Points — very important for exams

Standard (min/max) vs algebraic (product/sum) — compute both.

Triangular α -cut: linear interpolation formula.

Hedges: square, root, complement — compute at points.

Method marks: formula → substitution → answer.

Review Questions

- ? $A \cap \blacksquare$ at $x=2$ for the given sets?
- ? $A_{0.5}$ of $(0, 10, 20)$?

Section Summary

Operations and cuts are the course's arithmetic: formula-first, compute cleanly, and watch the classical laws fail gracefully.

Exam Pointers

- Compute standard and algebraic operations.
- Find α -cuts and add fuzzy numbers.

Lecture 17: Solved: Relations and Inference Traces

Learning Objectives

- Compose relations
- Trace inferences

Problem 1 — composition: $R = [[0.9, 0.2], [0.4, 0.8]]$ ($X \times Y$), $S = [[0.7, 0.5], [0.3, 1.0]]$ ($Y \times Z$). $R \circ S$: $(1, x)$: $\max(\min(0.9, 0.7), \min(0.2, 0.3)) = \max(0.7, 0.2) = 0.7$; $(1, y)$: $\max(0.5, 0.2) = 0.5$; $(2, x)$: $\max(0.4, 0.3) = 0.4$; $(2, y)$: $\max(0.4, 0.8) = 0.8$. Result $[[0.7, 0.5], [0.4, 0.8]]$.

Problem 2 — inference trace (Mamdani): inputs: temperature $t=32^\circ\text{C}$, humidity $h=70\%$. Sets: Temp {Cool (20,25,30), Warm (25,30,35), Hot (30,35,40)}; fan speed {Slow, Medium, Fast} triangular on $[0, 100]$. At $t=32$: $\mu_{\text{Warm}} = 0.6$, $\mu_{\text{Hot}} = 0.4$. Rules: Warm $\rightarrow$ Medium fires 0.6; Hot $\rightarrow$ Fast fires 0.4. Aggregate max of clipped Medium (0.6) and Fast (0.4); centroid ≈ 62 — fan at 62%.

Problem 3 — Sugeno: rules: if x is A then $z=2x+1$ ($w=0.7$); if x is A then $z=x+5$ ($w=0.3$); $x=4$: $z = (0.7 \times 9 + 0.3 \times 9) / 1.0 = 9$. Exam technique: composition — row-by-column max-min, show each cell; inference — fuzzify, fire (min), aggregate (max), defuzzify — label each stage.

Key Definitions

Inference trace stages is defined as the four labeled stages markers check: fuzzify, fire, aggregate, defuzzify.

Examples

- $R \circ S = [[0.7, 0.5], [0.4, 0.8]]$: every cell shown.
- Fan 62%: the complete labeled trace.

Key Points — very important for exams

- Max-min: row $\times$ column, max of mins — show all cells.
- Mamdani: min-fire, max-aggregate, centroid.
- Sugeno: weighted average of rule functions.
- Label stages — markers check each one.

Review Questions

? Recompute R $\square$ S with max-product instead.

? Defuzzify by MOM for the fan example.

Section Summary

Composition and inference are the course's engines: compute cell-by-cell, trace stage-by-stage.

Exam Pointers

- Compute a max-min composition.
- Trace Mamdani and Sugeno inferences.

Lecture 18: Case Studies: Fuzzy vs Crisp

Learning Objectives

- Argue for fuzzy modeling
- Write the essay

Case 1 — crisp failure: a thermostat switching at exactly 25°C chatters (rapid on/off near the boundary) and ignores 'how far' — the fuzzy controller ramps smoothly, saving energy and wear. Case 2 — medical: 'high fever' as crisp $\geq 39^\circ\text{C}$ misses 38.9°C; fuzzy grading captures clinical reality — diagnosis is inherently graded.

Case 3 — finance: credit scoring with crisp cutoffs ('income > 50k = safe') creates cliff effects; fuzzy scoring grades risk continuously — fairer and more predictive. Case 4 — project management: 'about 3 weeks' as fuzzy (2,3,5) in PERT gives realistic schedules; crisp 3-week estimates systematically underrun (planning fallacy meets false precision).

The meta-lesson: crisp models pretend precision the world doesn't have; fuzzy models formalize the vagueness experts actually use. Exam essay: 'Discuss, with examples, where fuzzy modeling outperforms crisp modeling' — structure: thresholds vs grades, cliff effects, expert knowledge capture, and the four cases above.

Key Definitions

Cliff effects is defined as the artifact where tiny input changes across a crisp threshold cause large output jumps.

Examples

- Thermostat chatter: crisp boundary, real-world failure.
- Credit cliffs: 49.9k vs 50.1k treated as different worlds.

Key Points — very important for exams

Thermostats, diagnosis, credit, scheduling: grades beat thresholds.

Cliff effects: crisp boundaries, unjust jumps.

Fuzzy captures expert knowledge in words.

Essay structure: four cases + the meta-lesson.

Review Questions

- ? Find a cliff effect in admissions cutoffs.
- ? Is fuzzy always better? When not?

Section Summary

Where the world is graded, model it graded: four cases proving fuzziness is realism, not sloppiness.

Exam Pointers

- Give four cases where fuzzy beats crisp.
- What are cliff effects?

Lecture 19: Question Bank and Final Revision

Learning Objectives

- Drill the paper
- Revise systematically

Question bank: Q1. Define fuzzy set, support, core, α -cut — one line each. Q2. $A=\{(a,0.2),(b,0.9)\}$, $B=\{(a,0.5),(b,0.4)\}$: compute $A\cup B$, $A\cap B$ (standard and algebraic), ■. Q3. Prove De Morgan's laws for min/max. Q4. Triangular (1,4,7): find the 0.5-cut; add (0,2,4). Q5. Compose $R=[[0.6,0.3],[0.2,0.9]]$ with $S=[[0.8,0.1],[0.4,0.7]]$ (max-min).

Q6. Trace: error=-2 (Negative 0.6, Zero 0.3), rules Negative→Decrease, Zero→Maintain: compute the crisp control action by centroid (assume triangular outputs). Q7. Mamdani vs Sugeno: compare in a table. Q8. Fuzziness vs probability: two examples. Q9. FCM vs k-means: when does FCM win? Q10. ANFIS: what learns, and how?

Long Q: Design a fuzzy fan controller: universes, terms, 9-rule table, full hand-trace for given inputs, and defuzzification. Revision order: definitions → computations (do each type twice) → traces → comparisons → the design question. The paper rewards complete traces with labeled stages.

Key Definitions

Mark-bearing computations is defined as the five computation types that carry most marks: operations, cuts, composition, traces, defuzzification.

Examples

- Q5: $[[0.6,0.3],[0.52,0.63]]$ — verify by hand.
- Long Q: the 9-rule table as the answer's backbone.

Key Points — very important for exams

Ten questions + one design long-question.

Computations twice each — fluency is marks.

Labeled stages on every trace.

Revision order: definitions → computations → traces → design.

Review Questions

? Which type is slowest for you? Drill it.

? Write the 9-rule table from memory.

Section Summary

The fuzzy paper is won by fluent hands: drill the five computations until each is automatic.

Exam Pointers

→ Compose the given relations.

→ Design and trace a fuzzy controller.

Lecture 20: Fuzzy Clustering

Learning Objectives

- Explain FCM
- Compute one iteration

Fuzzy clustering: Fuzzy C-Means (FCM) assigns each point membership grades across clusters (not hard labels) — minimizing $\sum \sum u_{ij}^m \|x_i - c_j\|^2$. The fuzzifier m (usually 2) controls softness. Applications: customer segmentation (a customer is 0.7 'premium', 0.3 'standard'), image segmentation, gene-expression analysis. vs k-means: FCM wins when boundaries are genuinely graded; loses on speed and crisp interpretability.

Exam computation: two points, two clusters, one FCM iteration — update memberships then centroids.

Know the formulas: $u_{ij} = 1 / \sum (\|x_i - c_j\| / \|x_i - c_k\|)^{2/(m-1)}$. The pattern: distances → memberships → weighted centroids → repeat. Very important: explain why soft assignments suit overlapping real-world categories.

Key Definitions

Fuzzifier m is defined as the exponent controlling cluster softness in FCM — $m=1$ is k-means, $m \rightarrow \infty$ is total fuzz.

Examples

■ Segmentation: a shopper 0.7 premium / 0.3 standard — soft truth.

■ $m=2$: the standard choice — justify it.

Key Points — very important for exams

FCM: soft memberships minimizing weighted distances.

Fuzzifier m controls softness; $m=1 \rightarrow$ k-means.

Wins on graded boundaries; costs speed.

One hand-iteration: distances $\rightarrow$ memberships $\rightarrow$ centroids.

Review Questions

? Why not always use FCM?

? Segment three customers softly.

Section Summary

Clustering for a graded world: soft memberships where categories genuinely overlap.

Exam Pointers

$\rightarrow$ When does FCM beat k-means?

$\rightarrow$ Do one FCM iteration by hand.

Lecture 21: Neuro-Fuzzy Systems (ANFIS)

Learning Objectives

- Explain ANFIS
- Position it in AI

Adaptive Neuro-Fuzzy Inference Systems (ANFIS): the hybrid that learns — a neural-network structure implementing a Sugeno fuzzy system, with membership-function parameters tuned by gradient descent + least squares. Architecture: 5 layers (fuzzify $\rightarrow$ fire $\rightarrow$ normalize $\rightarrow$ consequent $\rightarrow$ aggregate). The promise: fuzzy interpretability with neural learning.

Training: hybrid learning — forward pass fixes premises, solves consequents by least squares; backward pass backpropagates error to tune membership functions. Applications: function approximation, control tuning, forecasting. Limitations: curse of dimensionality (rules explode with inputs), and learned sets may lose linguistic meaning — the interpretability paradox.

Exam position: ANFIS as the course's synthesis — fuzzy logic's knowledge representation plus neural networks' learning. Compare: expert systems (knowledge, no learning) vs neural nets (learning, no knowledge) vs ANFIS (both). The future direction: neuro-fuzzy as the template for explainable AI.

Key Definitions

ANFIS layers is defined as ANFIS's 5-layer structure: fuzzify, fire, normalize, consequent, aggregate.

Examples

- Forward pass: least squares solves linear consequents — fast.
- Rule explosion: 5 inputs $\times$ 3 terms = 243 rules — the dimensionality curse.

Key Points — very important for exams

ANFIS: Sugeno system in a neural structure — it learns.

Hybrid training: least squares + backpropagation.

5 layers; applications in control and forecasting.

Paradox: learning can destroy linguistic meaning.

Review Questions

- ? Why does ANFIS need hybrid learning?
- ? ANFIS vs pure neural net: trade-offs.

Section Summary

The course's synthesis: fuzzy knowledge that learns — ANFIS as the template for explainable AI.

Exam Pointers

- How does ANFIS learn?
- What is the interpretability paradox?

Lecture 22: Fuzzy Time Series Forecasting

Learning Objectives

- Explain fuzzy forecasting
- Compare with statistics

Fuzzy time series and forecasting: modeling 'sales are high and rising' directly — Song & Chissom's framework: define the universe, partition into fuzzy sets, fuzzify history, build fuzzy logical relationships (FLRs), forecast by defuzzifying. Used in enrollment, stock-index, and energy-load forecasting where data is short and experts speak in words.

Worked sketch: university enrollments 1971–1992 (the classic dataset): universe [13000, 20000], 7 fuzzy sets ($A_1 \dots A_7$), FLRs like $A_i \rightarrow A_j$, $A_i \rightarrow A_k$; forecast = midpoint of consequent's centroid. The lesson: respectable forecasts from tiny data + linguistic structure — where statistics starves, fuzziness feeds.

Exam essay: 'Compare fuzzy time series with ARIMA' — structure: data needs (small vs large), assumption load (none vs stationarity), interpretability (linguistic vs coefficients), and accuracy (competitive on short series). The meta-point: fuzzy methods shine where data is scarce and expertise is verbal.

Key Definitions

Fuzzy logical relationships is defined as fuzzified temporal patterns like $A_{t-1} \rightarrow A_t$ — the fuzzy analogue of Markov transitions.

Examples

- Enrollments: 7 fuzzy sets forecasting from 22 data points.
- FLR $A_{t-1} \rightarrow A_t$, $A_t \rightarrow A_{t+1}$: the fuzzified transition table.

Key Points — very important for exams

Song & Chissom: fuzzify history, build FLRs, defuzzify forecasts.

Classic: university enrollments from 22 points.

Shines where data is scarce and expertise is verbal.

vs ARIMA: fewer assumptions, linguistic, competitive on short series.

Review Questions

- ? Forecast one step from given FLRs.
- ? When would you choose ARIMA instead?

Section Summary

Forecasting in words: fuzzy time series turns scarce data and expert language into predictions.

Exam Pointers

- How does fuzzy time-series forecasting work?
- Compare with ARIMA.

Glossary of Key Terms

Fuzzy set — a set with graded membership: $\mu_A(x) \in [0,1]$ for each element x .

Linguistic variable — a variable taking linguistic values (words), each modeled as a fuzzy set.

Hedges — modifiers like 'very' (concentration, μ^2) that reshape membership functions.

t-norms — binary operations generalizing AND (min, product) to fuzzy sets.

Max-min composition — $(R \circ S)(x,z) = \max \text{ over } y \text{ of } \min(\mu_R(x,y), \mu_S(y,z))$ — chaining fuzzy relations.

α -cut — the crisp set of elements with membership $\geq \alpha$ — the discretization tool for fuzzy arithmetic.

Fuzzification — converting a crisp measurement into membership grades in input fuzzy sets.

Defuzzification — converting the aggregated fuzzy output into a single crisp number (e.g., centroid).

Centroid — the center-of-gravity of the output fuzzy set — the standard defuzzification method.

Fuzzy C-Means — clustering where points hold membership grades across clusters — Bezdek's Fuzzy C-Means.

ANFIS — a Sugeno FIS as a neural network, learning membership parameters from data.

Generalized modus ponens — Zadeh's inference pattern: partial matching of premises yields partially-matched conclusions.

Rule table — the expert-knowledge table mapping input-term combinations to output terms.

Fuzzy queries — database queries with vague predicates evaluated by membership grades and ranked by satisfaction.

Choquet integral — aggregation with respect to a fuzzy measure — capturing criteria interaction (redundancy, synergy).

Fuzzy poverty measures — poverty measurement by membership grades rather than a binary poverty line.

Revision checklist — the exam-ready toolkit: definitions, five computation types, proof outlines, comparison tables.

Triangular α -cut formula — the α -cut of triangular (a,b,c) : $[a+\alpha(b-a), c-\alpha(c-a)]$ — linear interpolation.

Inference trace stages — the four labeled stages markers check: fuzzify, fire, aggregate, defuzzify.

Cliff effects — the artifact where tiny input changes across a crisp threshold cause large output jumps.

Mark-bearing computations — the five computation types that carry most marks: operations, cuts, composition, traces, defuzzification.

Fuzzifier m — the exponent controlling cluster softness in FCM — $m=1$ is k-means, $m \rightarrow \infty$ is total fuzz.

ANFIS layers — ANFIS's 5-layer structure: fuzzify, fire, normalize, consequent, aggregate.

Fuzzy logical relationships — fuzzified temporal patterns like $A \rightarrow A$ — the fuzzy analogue of Markov transitions.