# 📘 AI Notes – Vector Databases (pgvector)

This chapter explains Vector Databases in detail, focusing on pgvector. It is written for both deep understanding and interview preparation. Vector databases are critical in AI systems for semantic search and RAG pipelines.

## 1. What is a Vector Database?

A vector database stores embeddings (lists of numbers that represent meanings). Instead of keyword matching, it allows semantic search – finding text that has a similar meaning. Think of it as a Google Maps for meanings, where each sentence or document chunk is a point in semantic space.

## 2. Why Not Just Store Text?

Plain text search only works if exact words appear. For example, 'homeowners insurance' will not match 'house coverage'. Vector search solves this by comparing meanings, not words. This is why embeddings + vector DBs are the backbone of semantic search.

## 3. pgvector – Vector Search in PostgreSQL

pgvector is a PostgreSQL extension that adds a new column type: vector(N). N is the number of dimensions (e.g., 1536 for OpenAI embeddings). It supports similarity search operators:

• <-> : Cosine distance  
• <#> : Inner product  
• <=> : Euclidean distance

Example usage:

ALTER TABLE chunks ADD COLUMN embedding vector(1536);  
  
INSERT INTO chunks (id, content, embedding)  
VALUES ('1', 'Flood insurance covers rising water damage', '[0.12, -0.88, 0.53, ...]');  
  
SELECT id, content FROM chunks  
ORDER BY embedding <-> '[0.10, -0.87, 0.52, ...]'  
LIMIT 3;

## 4. Analogy – Library with GPS

A normal library index only lets you search by exact title words. A vector database assigns every book a GPS coordinate of meaning. When you ask 'books about space exploration', it finds nearby coordinates, not just exact matches.

## 5. Insurance Example

Carrier rules stored as embeddings:  
1. 'Water backup is excluded unless endorsed.'  
2. 'Flood damage is not covered under homeowners policies.'  
3. 'Homeowners insurance covers fire and theft.'  
  
Query: 'sewer backup coverage'  
• Plain SQL: No match (different words)  
• Vector DB: Finds #1 because meanings are close  
This shows why vector search is so valuable in insurance underwriting knowledge bases.

## 6. How Does Search Work?

Search involves three steps:  
1. Embed the query (convert to vector)  
2. Compare with stored embeddings using similarity metrics  
3. Return the nearest neighbors  
  
Similarity metrics:  
• Cosine similarity (angle between vectors, most common)  
• Dot product (alignment)  
• Euclidean distance (straight-line distance)

## 7. Scaling & Indexing

For small datasets (<100k vectors), brute force search is fine. But for millions of vectors, it becomes too slow. Vector DBs use Approximate Nearest Neighbor (ANN) algorithms to scale.

Key ANN algorithms:

• IVF Flat (Inverted File): Clusters vectors into buckets. At query time, search only in the nearest bucket.  
 Analogy: Instead of searching the whole US for a house, first jump into the right state, then city.  
  
• HNSW (Hierarchical Navigable Small World): Builds a graph of vectors with multiple layers. Search starts at a high-level graph and zooms into fine-grained neighbors.  
 Analogy: First use highways to get to the right city, then local roads to find the house.  
  
• PQ (Product Quantization): Compresses embeddings into smaller codes. This saves memory and speeds up search, at the cost of a little accuracy.  
 Analogy: Store thumbnails of images instead of full HD – good enough to recognize similarities.

pgvector currently supports IVFFlat indexes:  
CREATE INDEX ON chunks USING ivfflat (embedding vector\_cosine\_ops) WITH (lists = 100);

## 8. Popular Vector Databases

• pgvector (Postgres extension)  
• Pinecone  
• Weaviate  
• Milvus  
• FAISS (Meta library)

## 9. Professional Use Cases

• Semantic search (find relevant clauses in policies)  
• RAG pipelines (retrieve chunks for LLMs)  
• Deduplication (detect near-duplicate policies)  
• Clustering (group similar claims)  
• Recommendations (suggest similar policies)

## 10. Interview Prep – Common Questions

Q1: What is a vector database?

A: A DB optimized for storing and searching embeddings. Enables semantic search and similarity queries.

Q2: Why not just use keywords?

A: Keywords only match exact words. Vector DBs capture semantics, so 'auto policy' ≈ 'car insurance'.

Q3: What are common distance metrics?

A: Cosine similarity, inner product, Euclidean distance.

Q4: How do vector DBs scale?

A: They use Approximate Nearest Neighbor algorithms like IVF, HNSW, or PQ. These trade exact accuracy for massive speed gains.

Q5: What does pgvector support?

A: pgvector supports IVFFlat indexes for faster approximate searches.

✅ Key Takeaway: Vector databases turn embeddings into a searchable space of meanings. They are the backbone of modern AI search and RAG pipelines. For insurance-tech, they unlock powerful policy search, underwriting automation, and knowledge agents.