# Vector Databases

## Pinecone

Pinecone is a fully managed vector database designed specifically for high-performance similarity search and real-time retrieval of vector embeddings (e.g., from text, images, audio). It is cloud-native and focuses on scalability, speed, and simplicity.

**Key Features:**

- Fully managed service: No need to manage infrastructure or scaling.
- Real-time indexing and querying: Low-latency retrieval (ms scale).
- Namespace support: Logical separation of data.
- Metadata filtering: Combine vector similarity with metadata constraints.
- Automatic vector indexing: No manual index building.
- Scalable & distributed: Horizontal scaling with replication.
- Supports sparse-dense hybrid search.
- REST API & SDKs: Python, JavaScript, etc.
- Integrates with OpenAI, Cohere, LangChain, etc.

**When to Use:**

- You want a production-ready, cloud-native vector DB with minimal setup.
- You need real-time vector search with low latency.
- You want automated index management and horizontal scaling.
- You're building an AI product (chatbot, semantic search) and want plug-and-play infrastructure.

**Use Cases:**

- AI-powered search (semantic search, product search)

- Chatbots with retrieval-augmented generation (RAG)
- Recommendation systems
- Document similarity, image similarity

## Weaviate

Weaviate is an open-source vector search engine that includes a built-in graph-based database and automatic machine learning (ML) model integration. It's highly extensible and allows storing both vectors and rich object data (JSON).

**Key Features:**

- Open source with cloud and self-hosted options.
- Hybrid search: Combines keyword and vector-based search.
- Built-in modules: Integrations with OpenAI, Cohere, Hugging Face, etc.
- Custom vectors: Store and search vectors generated externally.
- Metadata filtering: Enables contextual filtering of results.
- Schema-based data model: Structured knowledge graphs.
- Multitenancy support.
- Horizontal scaling and sharding.

**When to Use:**

- You want a fully customizable and open-source solution.
- You're building a knowledge graph with semantic search.
- You need flexible schema support for rich metadata.
- You want tight integration with ML frameworks or models.

**Use Cases:**

- Semantic search for enterprise documents
- Recommendation engines
- Context-aware chatbots with knowledge graphs
- Cross-modal search (text → image, etc.)

## FAISS (Facebook AI Similarity Search)

FAISS is a library developed by Meta AI for efficient similarity search and clustering of dense vectors. It is not a full-fledged database but a library primarily focused on performance and local computation.

**Key Features:**

- Extremely fast vector search, optimized in C++ with Python bindings.
- Supports brute-force (exact) and approximate (ANN) search.
- Multiple indexing strategies (IVF, HNSW, PQ, OPQ).
- GPU acceleration for high performance.
- Local, in-memory storage (no persistent storage layer).
- No metadata support (pure vector search).
- Integration needed to combine with external metadata stores.

**When to Use:**

- You need high-performance local vector search.
- You're building a system where you control the infrastructure.
- You don't need metadata filtering or persistent storage.
- You're comfortable managing your own storage, scaling, and indexing.

**Use Cases:**

- Local prototyping of vector search systems
- Fast nearest-neighbor search for embeddings (e.g., image/text)

- Research and academic use
- Embedding clustering and analysis

## Azure AI Search

Azure AI Search is a cloud-based search-as-a-service offering from Microsoft Azure. It now supports vector search in addition to traditional full-text and filter-based search, allowing hybrid search experiences.

**Key Features:**

- Combines vector search with keyword search and filters.
- Native support for OpenAI embeddings and Azure OpenAI Service.
- Semantic ranking, synonym maps, and cognitive skills.
- REST API and .NET SDK integration.
- Built-in AI enrichment pipelines (e.g., OCR, language detection).
- Tight integration with Azure ecosystem (Blob storage, Cognitive Services).
- Indexing from external data sources (SQL, Cosmos DB, etc.).
- Security and access controls with Azure AD.

**When to Use:**

- You’re already in the Azure ecosystem.
- You need hybrid search (keyword + vector) for enterprise data.
- You want scalable, managed, secure search integrated with other Azure services.
- You’re building enterprise-grade apps with search + AI capabilities.

**Use Cases:**

- Enterprise knowledge search with hybrid capabilities
- Document intelligence platforms

- Secure internal document retrieval (HR, legal, IT)
- AI-powered enterprise apps (e.g., smart intranets)