🤖 MODEL INFERENCING FAQS

🔍 BASICS OF MODEL INFERENCING

1. What is model inferencing in machine learning?
Inferencing is the act of using a trained model to make predictions on unseen data.

2. How does inferencing differ from training?
Training updates model weights based on data; inferencing uses those fixed weights for predictions.

3. What are common applications of inferencing?
Image classification, voice assistants, translation, fraud detection, and recommendations.

4. What are the types of inferencing?

Real-time (low latency)

Batch (bulk processing)

Edge (on-device inference)

5. What is the role of inferencing in the AI lifecycle?
It is the final stage—where models are deployed to generate real-world predictions.

6. Why is inference latency important?
In latency-sensitive systems like chatbots or self-driving cars, delays can impact user experience or safety.

7. What is cold-start latency?
It's the initial delay in serving predictions when a model is first loaded.

8. How is inference throughput defined?
The number of predictions a model can serve per second (inferences/sec).

9. What is an inference engine?
A software system that executes trained models efficiently (e.g., ONNX Runtime, TensorRT).

10. How does model type affect inference behavior?
Outputs vary by model type: e.g., classifiers give labels, regressors give numbers, and transformers generate sequences.

⚙️ TECHNICAL COMPONENTS & CONCEPTS

11. What is batch size in inferencing?
The number of inputs processed together; larger batches increase throughput but may raise latency.

12. What is model quantization?
Reducing numeric precision (e.g., FP32 → INT8) to improve speed and reduce memory usage.

13. What are FP32, FP16, and INT8 formats?

FP32: 32-bit float (high precision)

FP16: 16-bit float (faster)

INT8: 8-bit integer (fastest, lowest precision)

14. Does reducing precision impact accuracy?
Yes, slightly—but often the trade-off is acceptable for performance gains.

15. What is synchronous vs asynchronous inference?

Synchronous: waits for prediction result

Asynchronous: runs inference in parallel with other tasks

16. What is model warm-up?
Pre-invoking a model with sample data to preload resources and reduce cold-start time.

17. Why is preprocessing consistency important?
Input data must match the training format for the model to make correct predictions.

18. What is a tensor in inference?
A multi-dimensional array holding model input or output data.

19. What does model compilation mean?
Transforming a model into an optimized form compatible with a specific backend or hardware.

20. What's the difference between inference latency and total latency?
Inference latency is model-only; total latency includes data prep and network transmission.

🧠 HARDWARE & SYSTEM CONSIDERATIONS

21. Should I run inference on a CPU or GPU?

CPU: cheaper, lower throughput

GPU: better for large/parallel workloads

22. What is a TPU?
Tensor Processing Unit—Google's custom hardware for accelerating ML workloads.

23. How does memory bandwidth affect performance?
Higher bandwidth enables faster data movement between memory and compute units.

24. What are edge devices in inferencing?
Devices like smartphones, Raspberry Pi, or IoT modules that perform inference without cloud dependency.

25. How do embedded systems perform inference?
Using compact, power-efficient models tailored to hardware constraints.

26. What is hardware acceleration?
Use of specialized chips (GPU, TPU, NPU) to speed up operations like matrix multiplication.

🛠️ OPTIMIZATION & EFFICIENCY TECHNIQUES

27. How can I optimize a model for inference?
Techniques include quantization, pruning, distillation, and batching.

28. What is model pruning?
Removing less useful weights to shrink the model and speed up execution.

29. What is model distillation?
Training a smaller "student" model to mimic a larger "teacher" model.

30. What is operator fusion?
Combining consecutive operations (e.g., conv + ReLU) to reduce overhead.

31. How do I choose the right inference backend?
Consider hardware, latency, throughput, and supported formats.

32. What are some common inference optimization tools?
ONNX Runtime, TensorRT, TVM, DeepSparse, OpenVINO.

33. How does ONNX improve inferencing?
It offers a standardized model format compatible with multiple tools and platforms.

34. What's the trade-off between performance and accuracy?
Performance boosts (via quantization/pruning) may slightly reduce accuracy.

📦 MODEL FORMATS & SERVING INFRASTRUCTURE

35. What are standard model formats for inference?
SavedModel (TF), TorchScript (PyTorch), ONNX, TensorRT Engine, OpenVINO IR.

36. What is TensorFlow Serving?
A serving system that provides REST/gRPC endpoints for TensorFlow models.

37. What is TorchServe?
PyTorch's serving framework that handles model inference with API support.

38. What is Triton Inference Server?
A high-performance NVIDIA server supporting multiple frameworks and GPUs.

39. What are inference APIs?
APIs expose your model over HTTP or gRPC to client applications.

40. How do I containerize a model?
Use Docker to package the model, its runtime, and dependencies into a single image.

41. What is an inference pipeline?
A sequence involving preprocessing → model execution → postprocessing.

🌐 DEPLOYMENT, MONITORING & SECURITY

42. How do I deploy a model to production?
Use serving tools (e.g., Triton, TensorFlow Serving), containerized with Docker and orchestrated via Kubernetes.

43. What is A/B testing in model serving?
Serving multiple model versions to compare performance or user impact.

44. How do I monitor inference endpoints?
Track latency, throughput, and errors using tools like Prometheus and Grafana.

45. What is autoscaling in inference?
Automatically adjusting the number of serving instances based on traffic.

46. How do I secure inference APIs?
Apply HTTPS, authentication, authorization, and rate limiting.

47. How do cloud platforms support inferencing?
AWS SageMaker, Azure ML, and GCP Vertex AI provide scalable, managed solutions.

48. How is mobile inference handled?
Using TensorFlow Lite or Core ML with lightweight models optimized for smartphones.

49. What are the best practices for validating inference?
Compare predictions to ground truth and monitor accuracy and drift over time.

50. How can inference performance be profiled?
Use built-in tools (e.g., NVIDIA Nsight, PyTorch Profiler) to analyze bottlenecks.