Machine Learning Engineer interview questions

This **Machine Learning Engineer** interview profile brings together a snapshot of what to look for in candidates with a balanced sample of suitable interview questions.

The **machine learning engineer role** is a highly technical role that is usually relevant to companies whose main product line has a very strong data-driven component. Machine learning engineers have the practical skills relevant to a data scientist but are particularly focused on the design and application of models build with machine learning to solve real world problems. As such, a machine learning engineer will have studied both the theoretical basis and the practical applications of machine learning and be particularly strong in related fields such as statistics, optimization, data mining and algorithmic design.

They know how to choose the right type of model for a particular problem having a diverse array of these at their disposal. For each model, they understand the limitations and assumptions, how to tune and improve model performance as well as use the right metrics to evaluate model accuracy. Research is often a core skill for this role and candidates with a strong research background such as having a PhD are highly sought after. From a practical perspective, candidates will have experience working with specialized tools and packages for machine learning such as scikit-learn (Python), Spark ML, R, Mahout and so on. Candidates will most often approach this role from a computer science or statistics background.

Role Specific Questions

(Understanding how a model works)

A fantastic way to start and build up a technical conversation is to have a candidate describe how a model with which they are familiar works. Technical interviews can often be very stressful for candidates and this is one way to allow candidates to relax slightly and talk about something in which they have more experience. It doesn't matter if they choose something very simple because the goal is to see if the candidate really understands the model and doesn't just know the basics. Going into substantial depth on something as simple as k-nearest neighbors or linear regression can be quite revealing about a candidate.

- What type of problem does the model try to solve?
- Is it prone to over-fitting? If so – what can be done about this?
- Does the model make any important assumptions about the data? When might these be unrealistic? How do we examine the data to test whether these assumptions are satisfied?

- Does the model have convergence problems? Does it have a random component or will the same training data always generate the same model? How do we deal with random effects in training?
- What types of data (numerical, categorical etc…) can the model handle?
- Can the model handle missing data? What could we do if we find missing fields in our data?
- How interpretable is the model?
- What alternative models might we use for the same type of problem that this one attempts to solve, and how does it compare to those?
- Can we update the model without retraining it from the beginning?
- How fast is prediction compared to other models? How fast is training compared to other models?
- Does the model have any meta-parameters and thus require tuning? How do we do this?

(Deeper machine learning questions)

- What is the EM algorithm? Give a couple of applications
- What is deep learning and what are some of the main characteristics that distinguish it from traditional machine learning
- What is linear in a generalized linear model?
- What is a probabilistic graphical model? What is the difference between Markov networks and Bayesian networks?
- Give an example of an application of non-negative matrix factorization
- On what type of ensemble technique is a random forest based? What particular limitation does it try to address?
- What methods for dimensionality reduction do you know and how do they compare with each other?
- What are some good ways for performing feature selection that do not involve exhaustive search?
- How would you evaluate the quality of the clusters that are generated by a run of K-means?

(Tools and research)

- Do you have any research experience in machine learning or a related field? Do you have any publications?
- What tools and environments have you used to train and assess models?
- Do you have experience with Spark ML or another platform for building machine learning models using very large datasets?