Twitter Thread by Santiago



Santiago
@svpino

10 machine learning questions you should be able to answer.

I started experimenting with a questionnaire before interviewing candidates. Applicants submit a video or written answers to apply for the position.

Thread: Let's look at 10 of these questions.

The questionnaire solves a couple of problems:

1. Weeds out candidates that aren't ready for the job.

2. It significantly cuts down the time we spend talking synchronously.

One problem: It requires extra time from the candidate before the phone interview.

Here are some numbers on a job posting before using the questionnaire:

• 249 applicants
• 13 good applications

That's only 5%.

This sucks.

Numbers after using the questionnaire:

• 44 applicants
• 21 good applications

That's 47%.

This is much better.

I look at it this way:

Sooner or later, I'll ask these questions, and the candidate will have to answer them.

I'd rather give them space and the opportunity to think about their answers without any pressure.

Those who can't or don't want to answer won't apply. That's fine.

I'm not worried about candidates searching for answers or asking someone else.

They will fool the questionnaire, not the interview that will come later.

Remember:

• The goal of the questionnaire and the resume is just to decide who moves to an interview.

Questions must be open-ended. No gotchas.

I want to allow candidates to think creatively and elaborate on their answers.

Preferred: A video response with the answers. Fallback is written answers if the candidate is not comfortable with video.

There are 10 questions on the questionnaire.

These cover basic principles that we want to ensure candidates understand.

Every candidate receives the same questions.

Let's go over them:

1. Provide a couple of examples where you used Supervised and Unsupervised machine learning techniques.

2. Walk us through the process you follow to select the best machine learning algorithm to solve a problem.

3. Explain the trade-off between bias and variance.

4. How do you decide how you should split your dataset when working in a supervised machine learning problem?

5. How do you handle imbalanced datasets in a classification problem?

6. Explain how you would approach reducing overfitting on a model.

7. Explain three different problems where using accuracy, precision, and recall is the best metric, respectively.

8. What would you expect to see as you vary the batch size when training a neural network?

9. What would you expect to see as you vary the learning rate when training a neural network?

10. What are the advantages of a Convolutional Neural Network over a fully connected network for image classification?

I post threads like this every week. You can find them here: @svpino.

Follow along for a good bunch of practical tips and epic stories about my experience with machine learning.