Start recording …

Keep-away Soccer!

https://www.cs.utexas.edu/~AustinVilla/sim/keepaway/

Admin

- **Draft due March 24th**
- Session moderators for today: **Taghian Jazi, Mehran**
    - https://docs.google.com/spreadsheets/d/1dbmlvduupZUCDjxU4HW2_350OVrVG-g1FoEAG-uWhMk
- Work on your projects should be well underway!
    - Have you run your first experiments yet?
- A complete draft is due next month:
    - Including results from your first experiment
    - Completed text, no typos, etc

Today’s Plan

- Talk about expectations for presentations
- More on the data of RL and statistical tools
- Project standups
- Your questions (including live ones via zoom)

Presentations start Wednesday: what to expect

The plan

- Each presentation will be 20 to 30 mins long
  - Each speaker gets a 40 min slot
- You can prerecord your presentation and I can play the video during lecture -OR- you can do it live
- If you choose to do it live, please assume there will be questions (ie part of the 20/30 mins)
- After each presentation we shift to advice & improvement phase:
  - Help the speaker with ideas for improving the presentation
  - More importantly: help the speaker with ideas for improving their project

Mark breakdown

- Polish (20%):
  - Little to no typos or grammar errors
  - Clear and useful figures
  - Reasonable template, use of color, and emphasis
- Structure (10%):
  - Logical flow of ideas
  - Useful Outline
- Content (20%):
  - Idea / problem well motivated
  - Simple and clear

- Delivery (50%):
  - Did you follow the rules I laid out in lecture e.g,:
    - one plot per slide
    - explain the axis first
    - side titles are topic sentences
    - one idea per slide
    - etc…
  - Scan the rules I presented, if you break them it will cost you

Distillation of Adam’s presentation advice

- Assume the audience doesn’t know much
- Always be simple and direct: say things explicitly
- One to two main ideas
- Warm up with title slide
- Use outline throughout (no meaningless words)
- Spend proper time to motivate
- Jargon and notation budgets
- Checkin with audience often
- Advice on algorithms and code
- Empirical results: slow & one thing at a time
- Rules showing data
- Have a conclusion! Talk about limitations
- All of “Low-level advice” slide

Back to the data …

Think of the typical RL experiment loop

- We have agent A and agent B (with all the hyper-parameters set, somehow…)
- We have our environment
- We run agent A on the environment, then agent B on the environment
    - Perhaps we compute the average return per episode -> for M episodes we get one number
    - We do this N times for agent A and agent B
- This process gives us N scalar numbers for each agent
- To make it concrete we have: N=30, M=200, agent A = Sarsa, agent B = Q-learning, and environment is Mountain Car

What do we do with the Data?

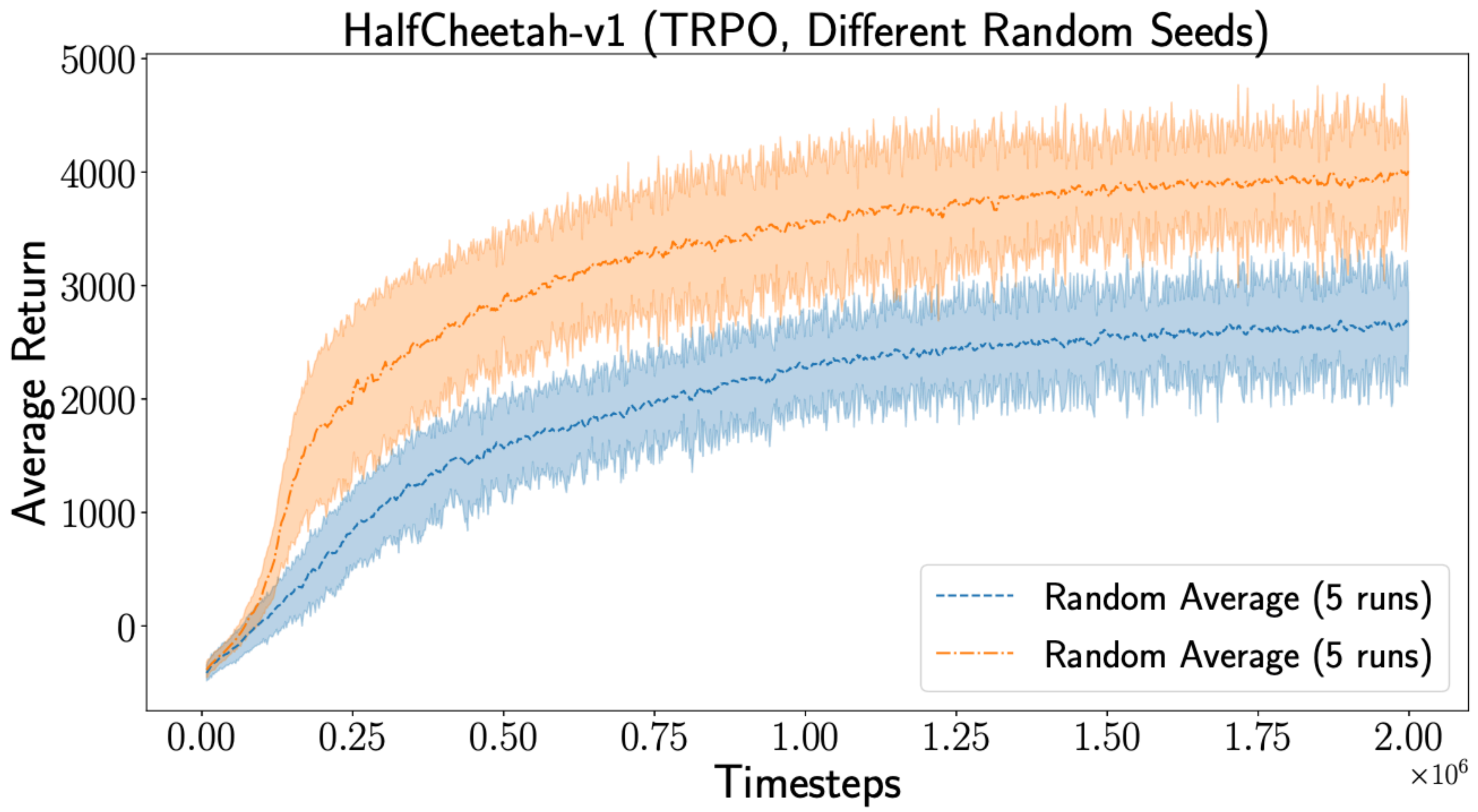
- For Sarsa we compute the average (over the 30 runs) of the mean return per episode -> gives us one number
- For Sarsa we compute the standard deviation (over the 30 runs) of the mean return per episode -> gives us one number
- Then we can characterize the mean performance and standard error (standard_deviation / sqrt(30))
- Standard error bars assume normality of the data, so you should check if that is true

We can do the same with learning curves

- We took **average return per episode** (over 200 episodes) as the performance statistic (producing one # per run)
- We could have also stored **return per episode** instead (producing 200 #'s per run)
- Then averaging over runs, producing average learning curves and computing standard error bars

But how do we know if the result is significant?

- Do the the average perf +/- standard error overlap between Sarsa and Q-learning?
- Do the error bars of the average learning curves for Sarsa and Q-learning overlap?
    - We know that can sometimes be misleading
    - We have to be on guard about our assumptions
- We can do hypothesis tests



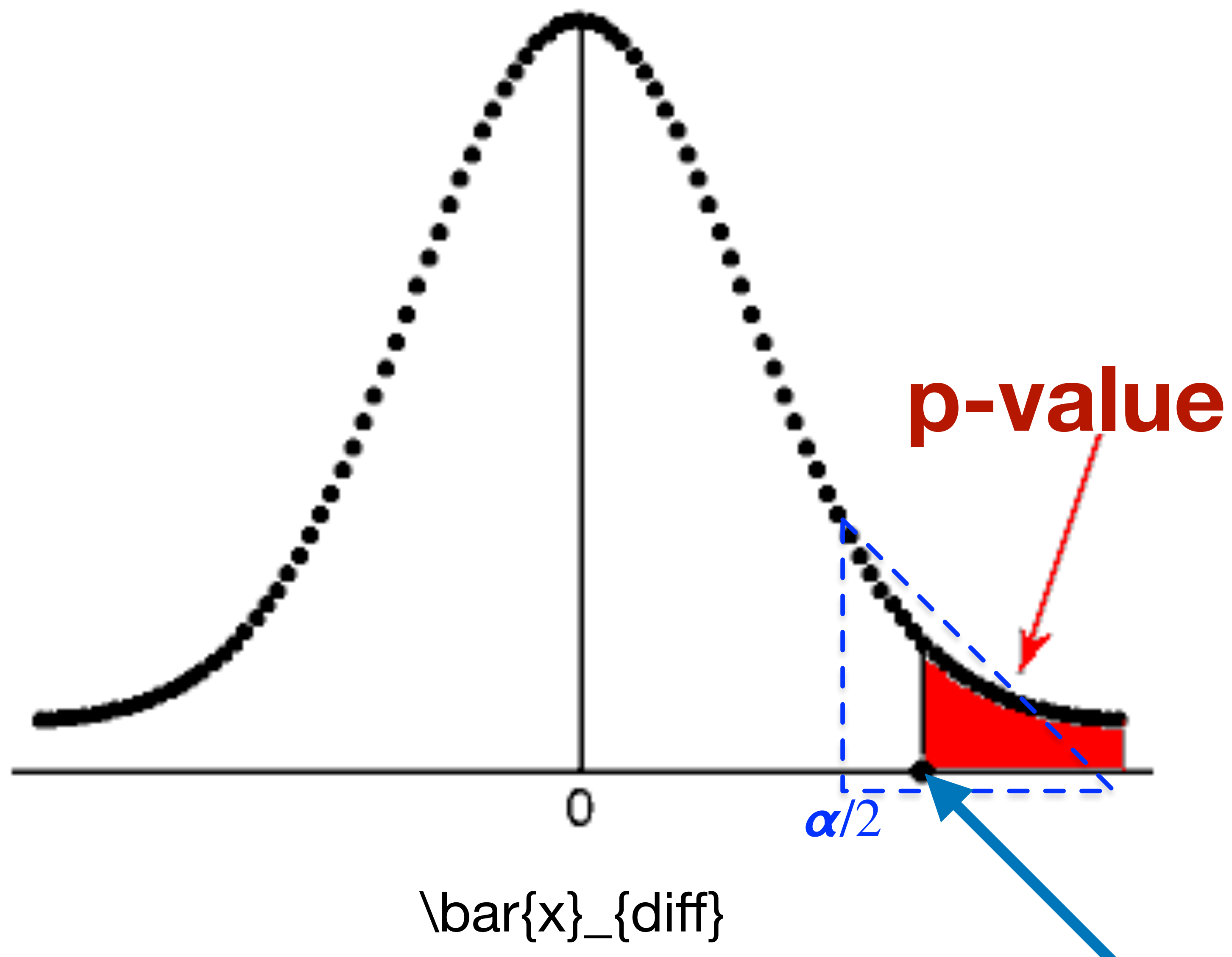
Comparing algorithms

- Imagine we ran Sarsa and Q-learning 30 times computing the mean episodic return over 200 episodes
    - We would have 30 pairs of numbers
- Take the difference between each of each pair and report the mean difference
    - We get one number: $\bar{x}_{diff}$
- Assume the true difference between the two is zero: **their perf is actually the same on mountain car**
- How does this relate to hypothesis testing and p-values

Assuming the null

- Assume the true difference between the two is zero: **their perf is actually the same on mountain car**

Perhaps $\bar{x}_{diff} >> 0$

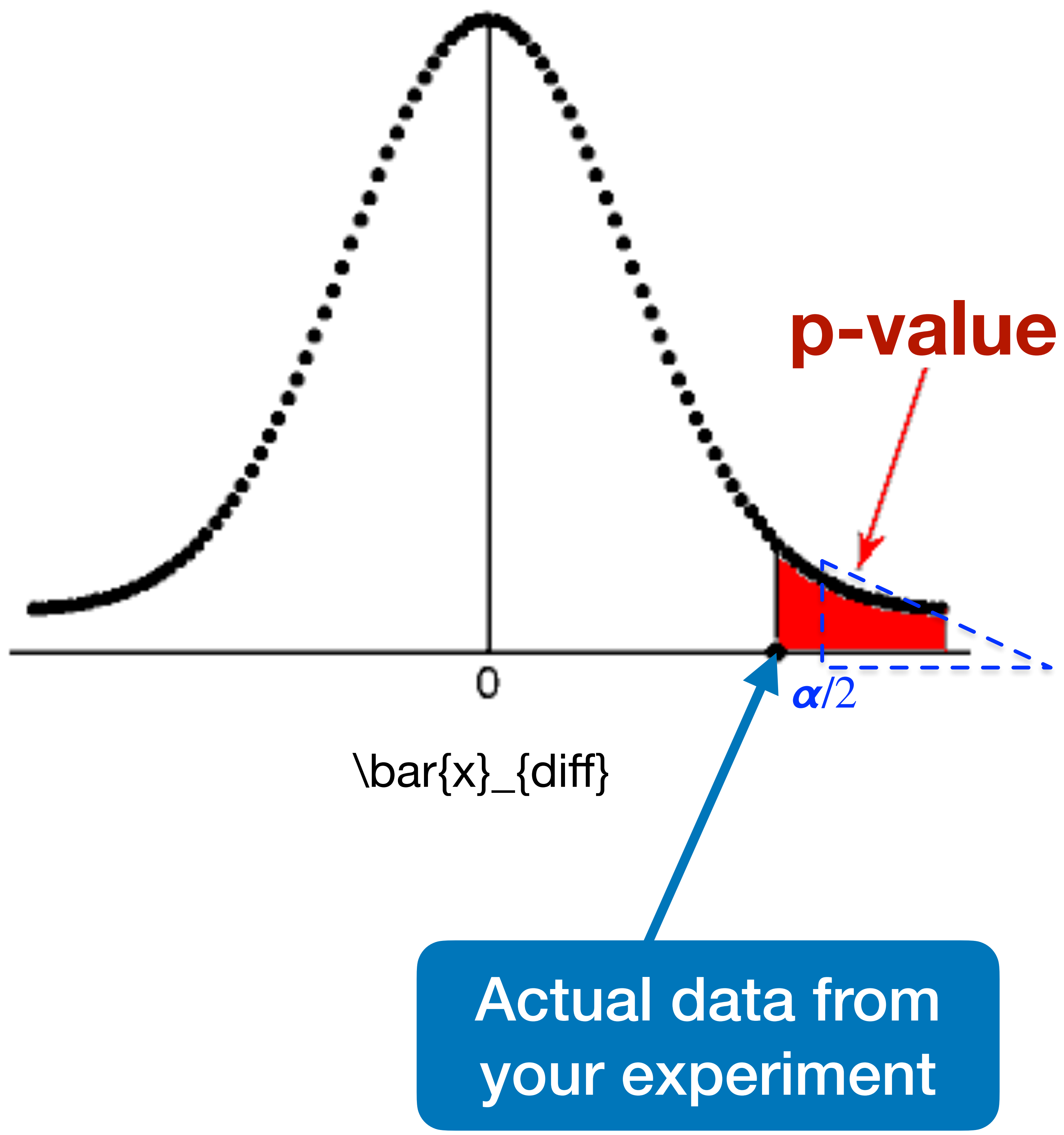


- **If** this probability is really small, **then** we **reject the null**
- We **declare** this assumed model p_{true} is likely **incorrect**
- We **declare** we have enough evidence that $\mathbb{E}[X_{diff}] \neq 0$

Assuming the null

- Assume the true difference between the two is zero: **their perf is actually the same on mountain car**

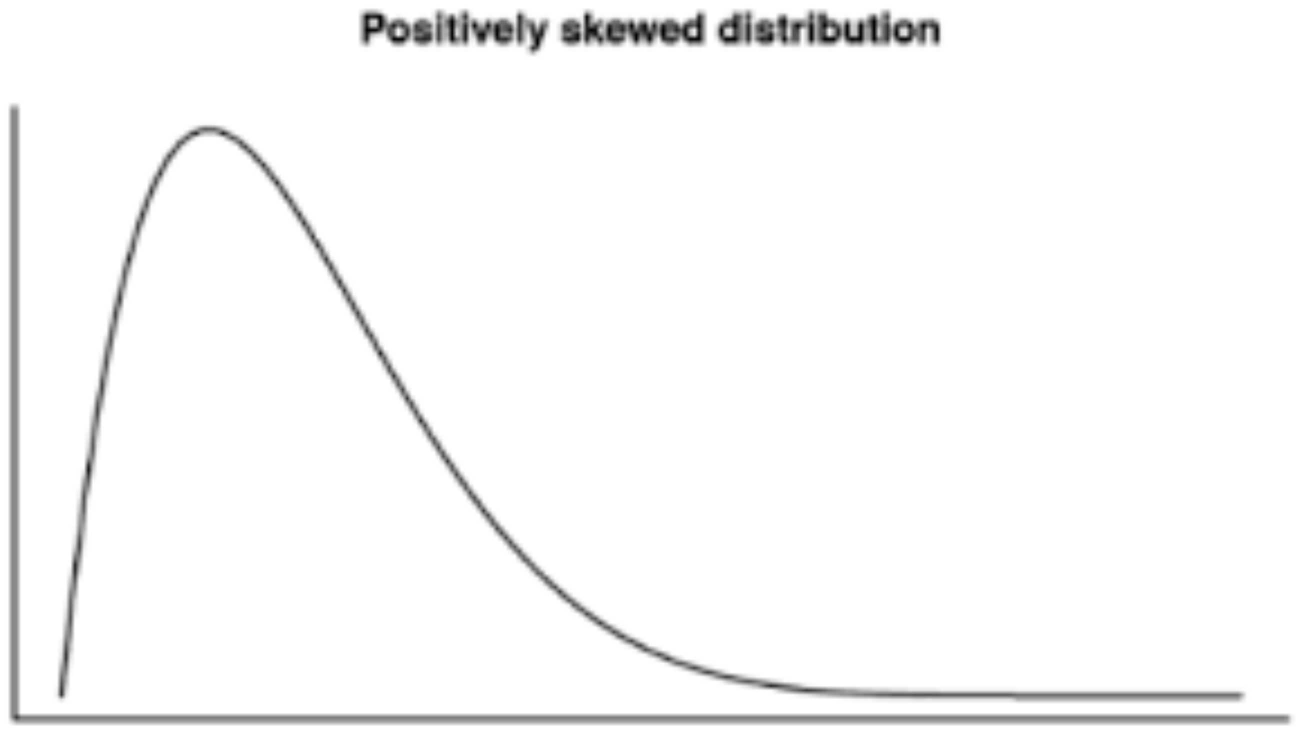
Perhaps x_{diff} is close to 0

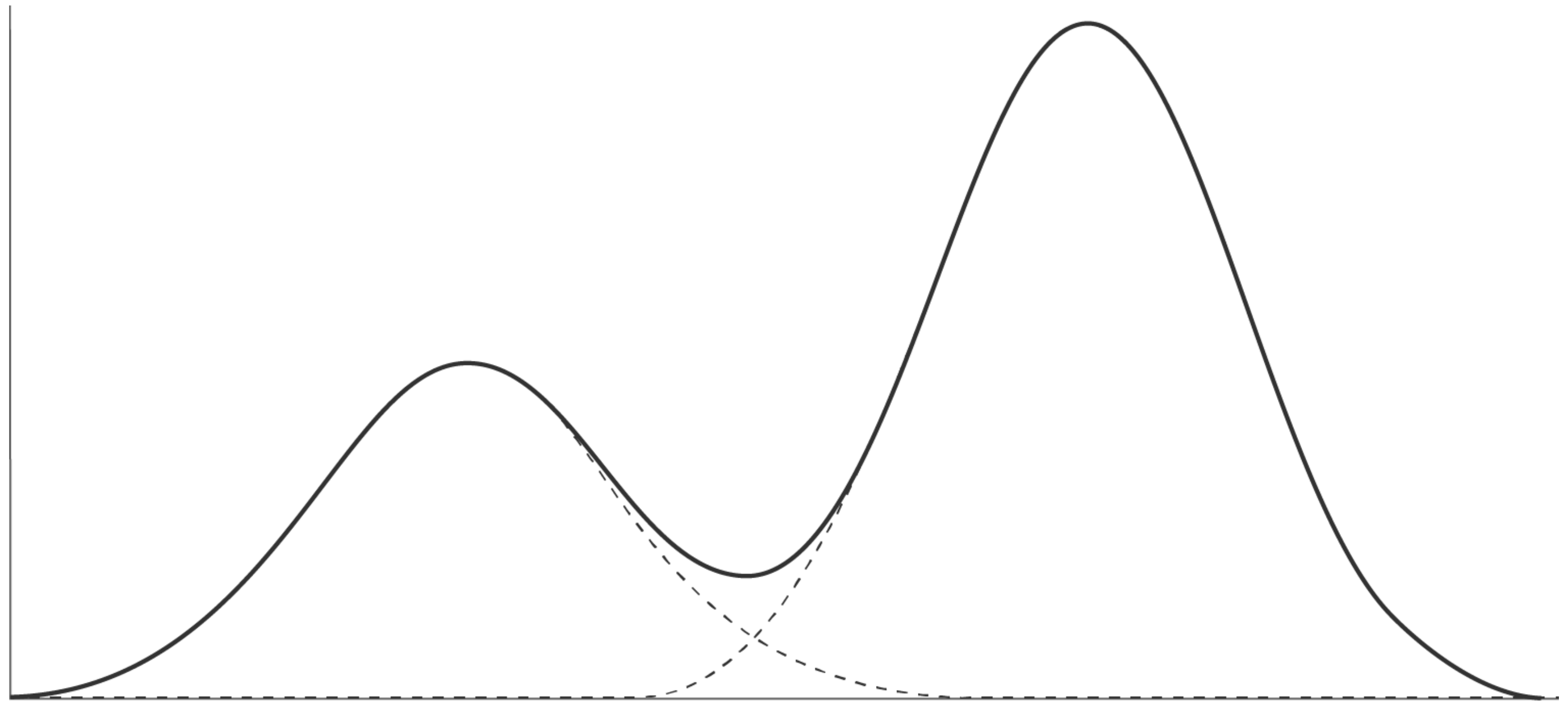


- **If** this probability is larger than some $\alpha/2$, **then** we **fail to reject the null**
- We can't say model p_{true} is **incorrect**
- We **declare** we have insufficient evidence that $\mathbb{E}[X_{diff}] = 0$ is not true

Where do the assumptions come in

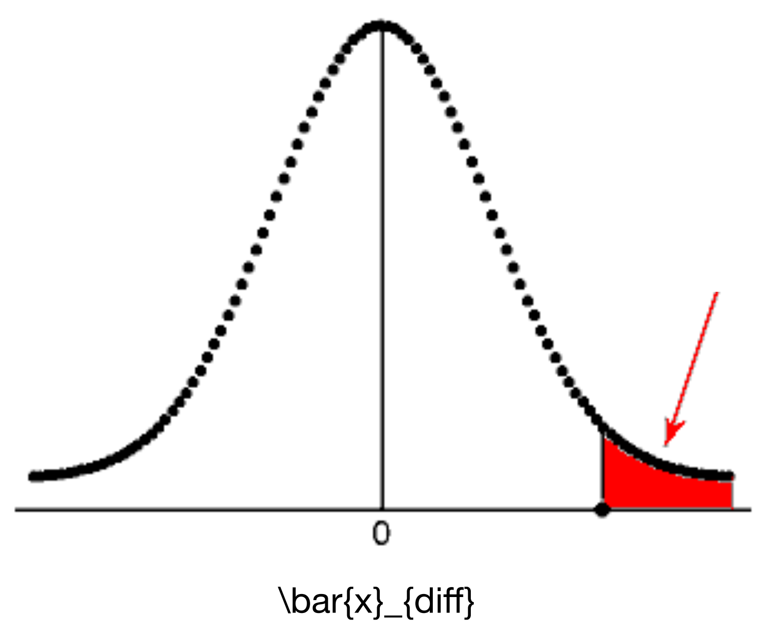
- How did we compute the p-value?
  - Some equation from a statistics textbook?
- We assumed that p_{true} was a gaussian
  - The true distribution of differences could be very different
  - Maybe the p-value would be incorrect





We might reject the null when the perf is actually the same!

- Even if we know, for sure, that the true distribution of differences is normal: $\bar{x}_{diff} \sim p_{true}$
    - There is a tiny probability that we would incorrectly reject the null
    - and say they are different
- This is called a **Type 1 Error**: *false positive*
- The significance level of the test α (the thing we compare the p-value with) is the probability of a **Type 1 Error**
    - The probability that we observe an $\bar{x}_{diff}$ far from zero, that is possible under p_{true} where p_{true} assumes $\mathbb{E}[X_{diff}] = 0$
- Statistical significance does not refer to the truth,
  but the probability of errors under our modelling assumptions



What if the assumptions don't fit the data?

- If p_{true} is skewed or bi-modal then the probability of incorrectly rejecting the null might be higher
  - Perhaps we decide to reject the the null, but we don't really have a valid basis to do so because the model was wrong
- We can empirically investigate these errors with synthetic data

The probability of Type 1 Errors

- Data sampled from two zero mean distributions: N samples (so N=#runs)
- Repeat the whole procedure 10^3 times and count the number of times we invalidly reject the null with different N and different tests (with different assumptions)

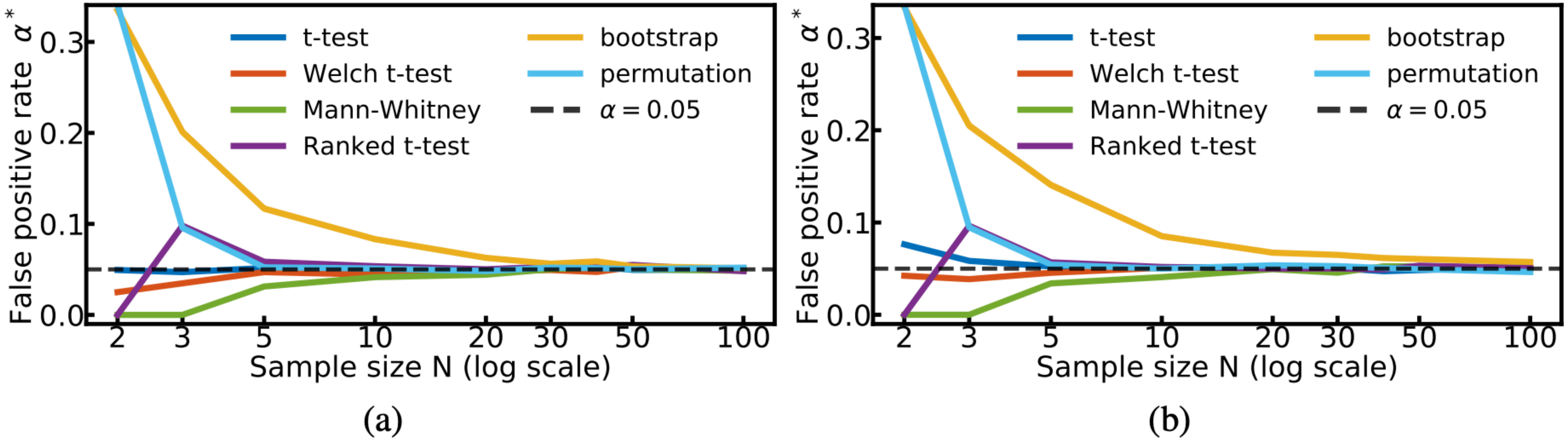
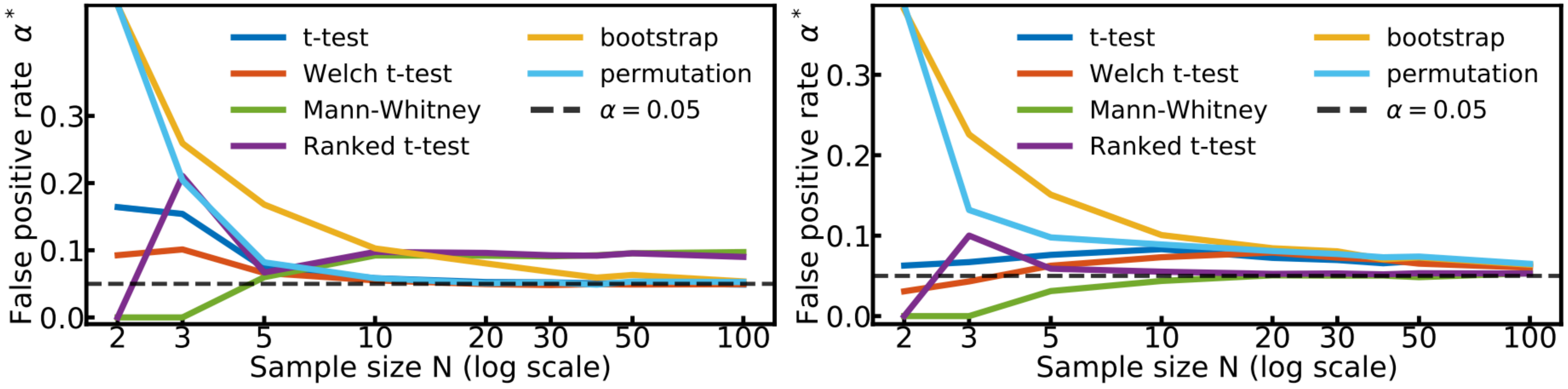


Figure 3: **False positive rates for same distributions, equal standard deviations.** Both samples are drawn from the same distribution ($\mu = 0$, $\sigma = 1$). **(a)**: A standard normal distribution. **(b)**: A bimodal distribution.

The probability of Type 1 Errors: unequal std deviations



(a) (b)

Figure 4: **False positive rates for same distributions, different standard deviations.** x_1 and x_2 are drawn from the same type of distribution, centered in 0 (mean or median), with σ_1 =1 and $\sigma_2 = 2$. **(a)**: Two bimodal distributions. **(b)**: Two log-normal distributions.

The probability of Type 1 Errors: “real” data

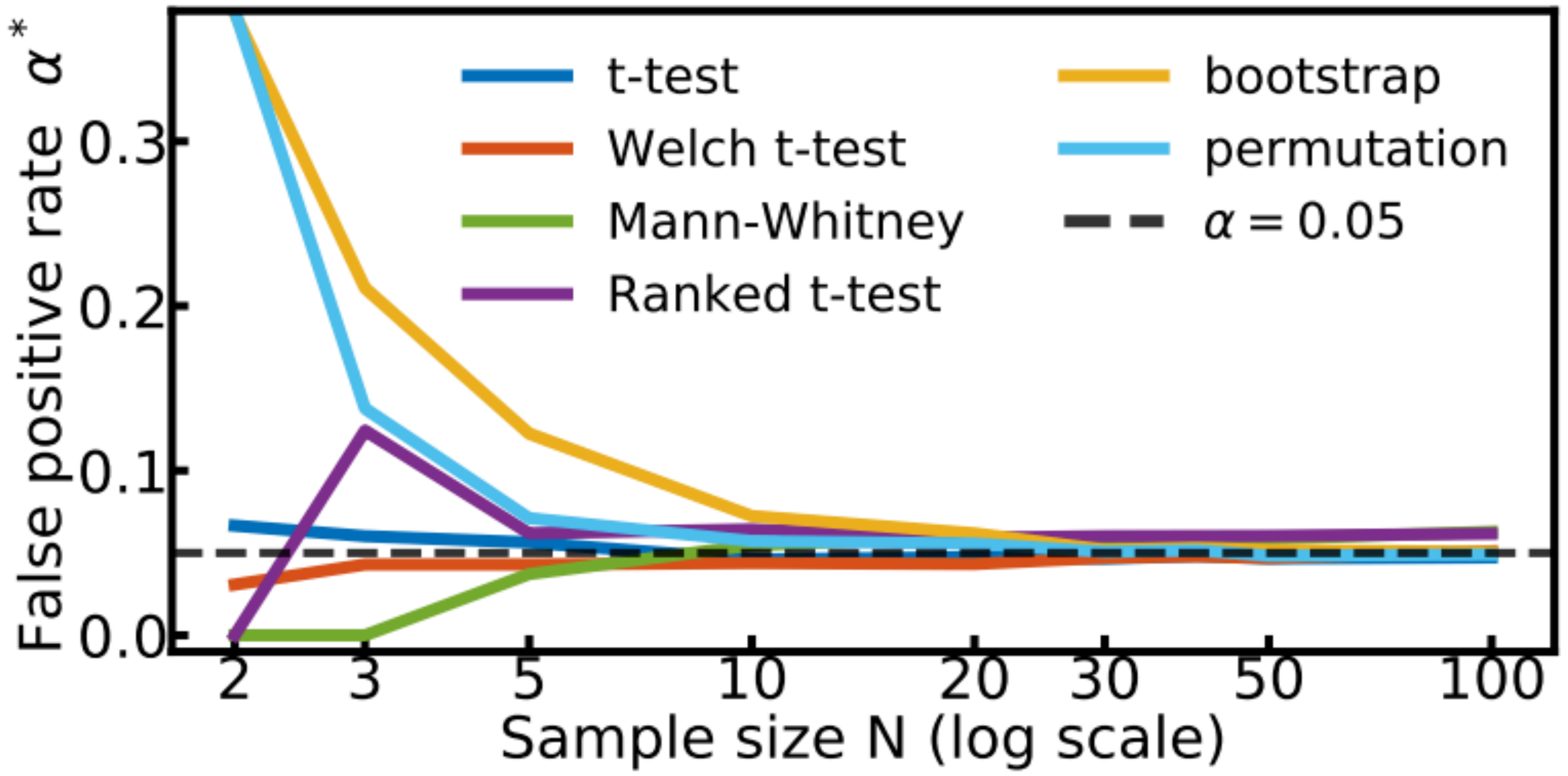


Figure 7: **False positive rates when comparing SAC and TD3.** x_1 is drawn from SAC performances, x_2 from TD3 performances. Both are centered in 0 (mean or median), with $\sigma_1 = 1.313$ and $\sigma_2 = 1.508$.

Stats wrap up

- We cannot always make α smaller (that is data & distribution dependent)
- We don't want to blindly increase the number of runs
- We can check the distribution of data produced by our agents and select the correct test:
  - T-test; variants of the T-test, Bootstrap, permutation, etc …
- All of this is relevant to confidence intervals and standard error bars
- There are other errors, like **falsely claiming the agent's perf is the same**
- **At the very least:** indicate the number of runs and type of variation method used
  - But we can do better …

References

- https://arxiv.org/pdf/1904.06979.pdf
- https://arxiv.org/pdf/1806.08295.pdf
- https://arxiv.org/abs/2002.05651

Project standup

- 30 second to 5 minute summary of your project
- Thing you are most focused on now
- Open question for the group:
    - Anything you are currently stuck on?

Your questions

- How do we measure learning rate?
  - Related how can we focus our experiments on speed of learning

Your questions

- How do reduce the amount of compute we need?
  - Besides smaller environments and less hyper-parameter sweeping

Your questions

- How do we do large parameter sweeps?
  - Dealing with lots of data
  - Looking at too many learning curves (visual inspection)
  - Pruning & local minima
  - Is this a way to select hypers or characterize performance?

Your questions

- Why don’t we see results in the literature with simple baseline policies?
    - Random agent, select the one best action
    - Are their other naive agents we could compare against?

Your questions

- How far do you think we are from having efficient RL algorithms that can be applied without tons of domain knowledge and tuning in the real world?

Your questions

- What do you think are the 5-10 year intermediate goal posts for RL?

Live questions

Your time is now!!