Joint meeting

Andrea Pierré

March 4th, 2024

Brown University

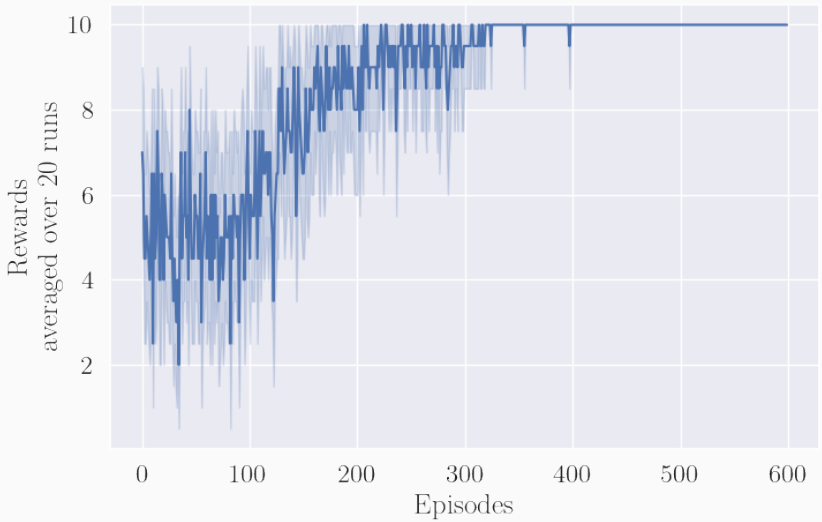
Outline

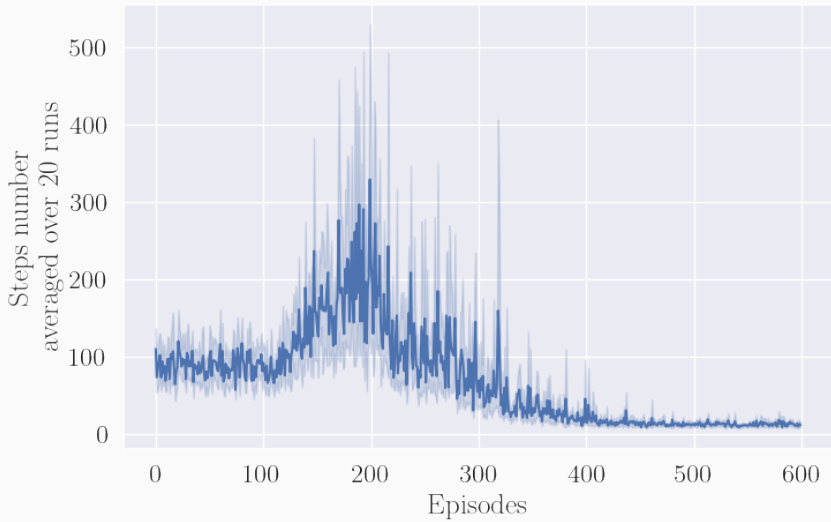
1. Online Deep RL training
2. Generalization experiment
3. Discussion

Outline

1. Online Deep RL training
2. Generalization experiment
3. Discussion

Rewards & steps





Why is it converging now?

- Lights cues in the state?
- Start training once replay buffer is full (5000 transitions) instead of when there are enough transitions for a batch (32 transitions)
- Soft update of the networks weights (instead of sharp transition)
- Huber loss instead of mean squared error $\rightarrow$ should be less sensible to outliers
- Remove ReLU on output layer!

Why is it converging now?

- Lights cues in the state?
- Start training once replay buffer is full (5000 transitions) instead of when there are enough transitions for a batch (32 transitions)
- Soft update of the networks weights (instead of sharp transition)
- Huber loss instead of mean squared error $\rightarrow$ should be less sensible to outliers
- Remove ReLU on output layer!

Why is it converging now?

- Lights cues in the state?
- Start training once replay buffer is full (5000 transitions) instead of when there are enough transitions for a batch (32 transitions)
- Soft update of the networks weights (instead of sharp transition)
- Huber loss instead of mean squared error $\rightarrow$ should be less sensible to outliers
- Remove ReLU on output layer!

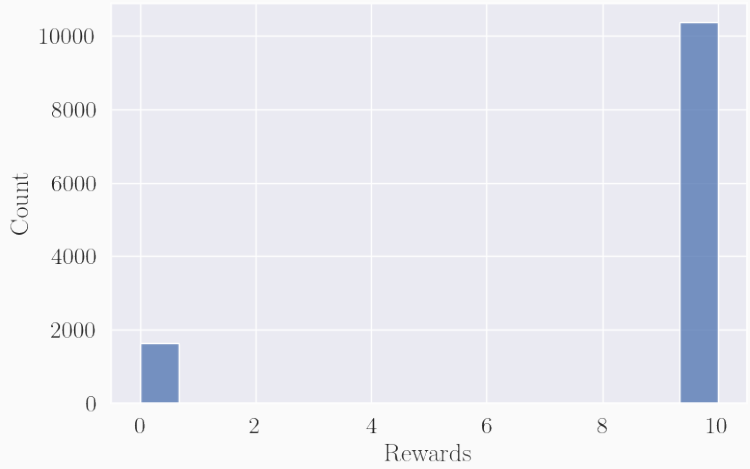
Why is it converging now?

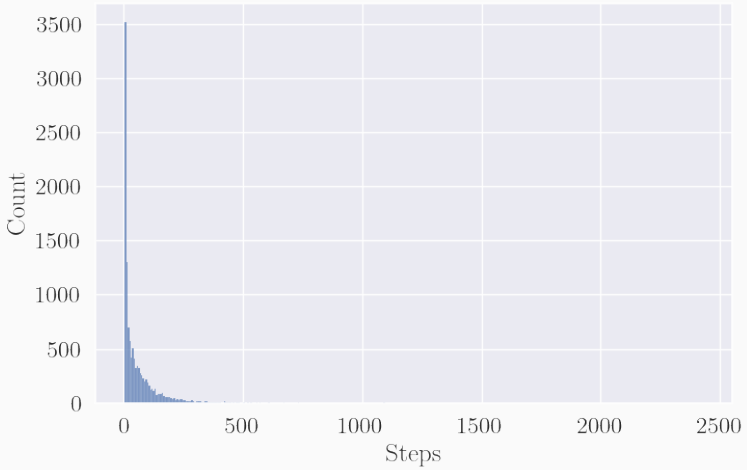
- Lights cues in the state?
- Start training once replay buffer is full (5000 transitions) instead of when there are enough transitions for a batch (32 transitions)
- Soft update of the networks weights (instead of sharp transition)
- Huber loss instead of mean squared error $\rightarrow$ should be less sensible to outliers
- Remove ReLU on output layer!

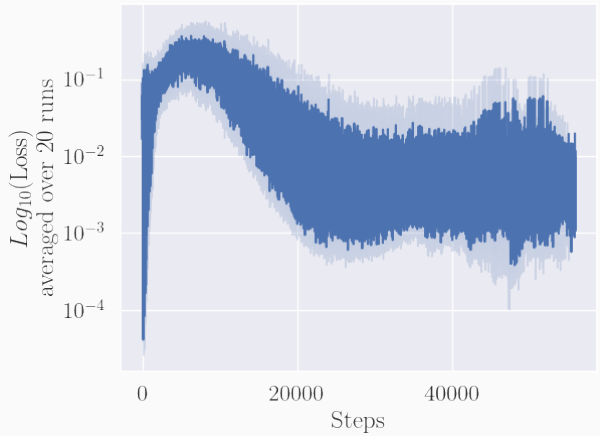
Why is it converging now?

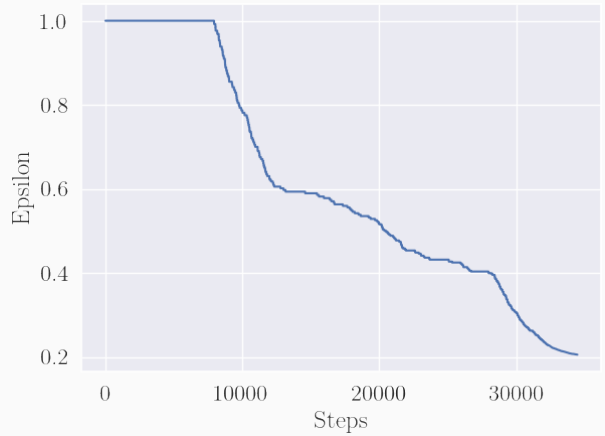
- Lights cues in the state?
- Start training once replay buffer is full (5000 transitions) instead of when there are enough transitions for a batch (32 transitions)
- Soft update of the networks weights (instead of sharp transition)
- Huber loss instead of mean squared error $\rightarrow$ should be less sensible to outliers
- Remove ReLU on output layer!

Loss, rewards & steps distributions, exploration/exploitation rate

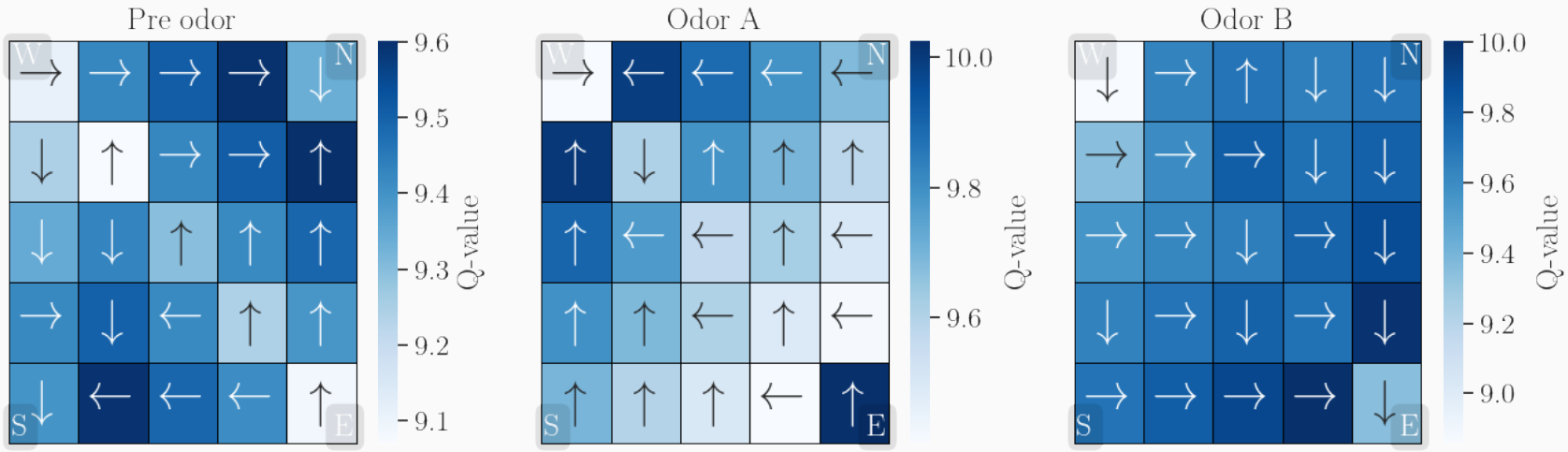








Policy learned



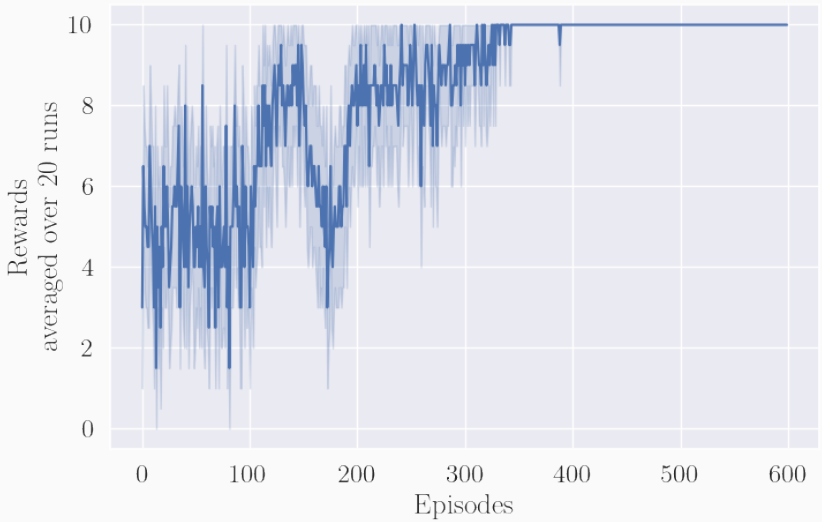
Outline

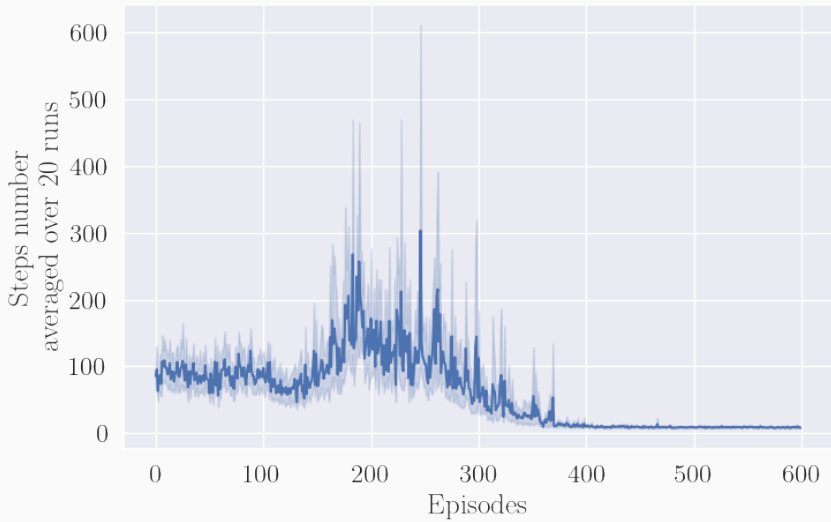
1. Online Deep RL training

2. Generalization experiment

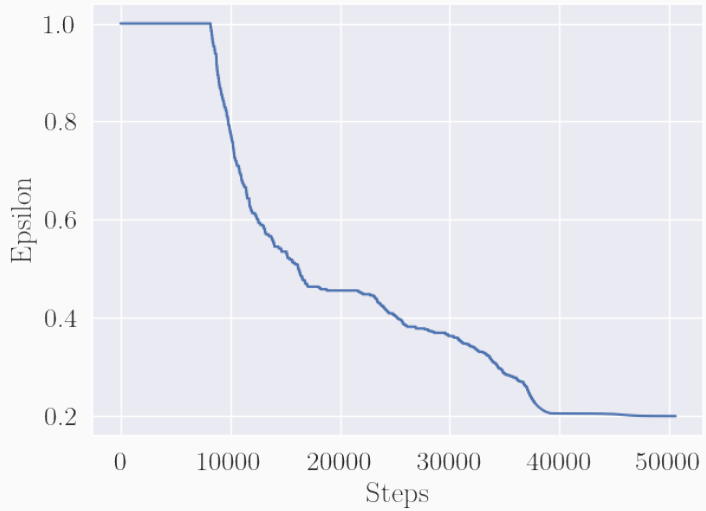
3. Discussion

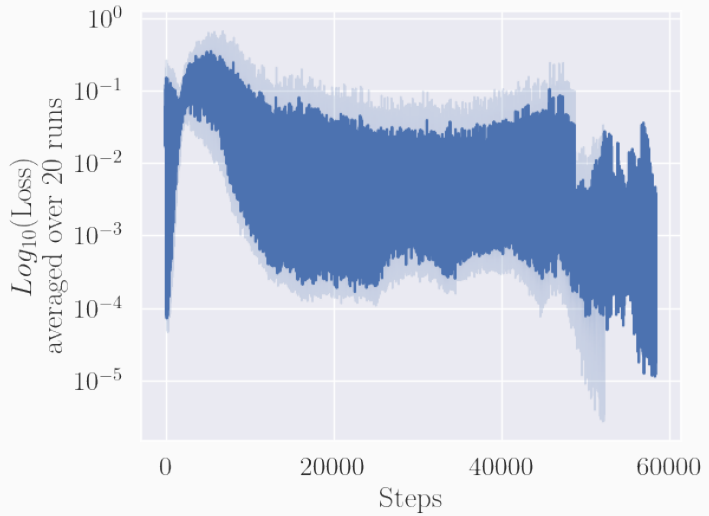
Training only in the lower triangle

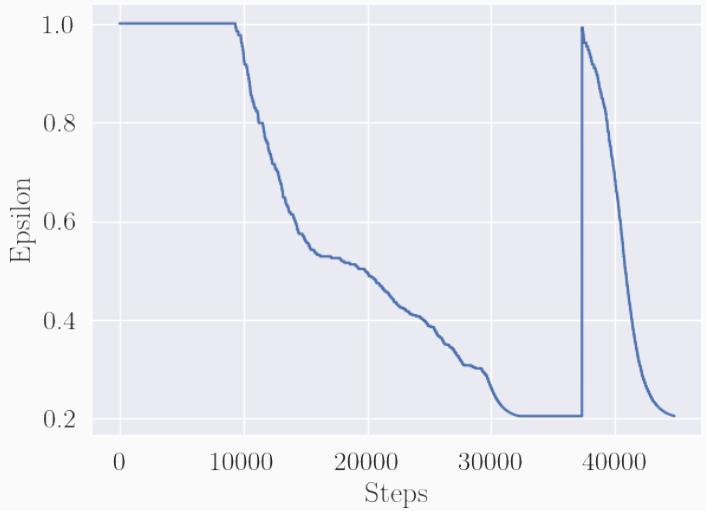


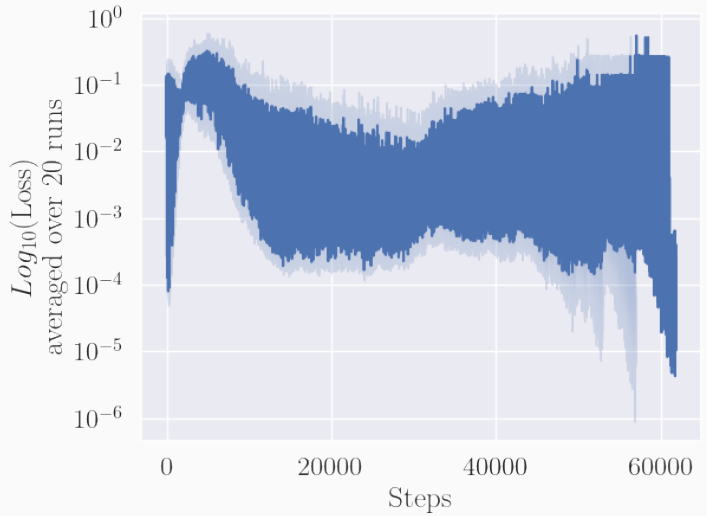


Training only in the lower triangle then switch to the upper triangle

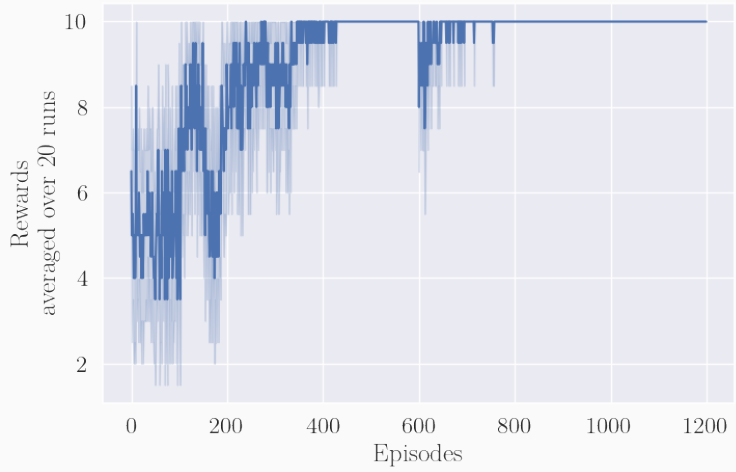


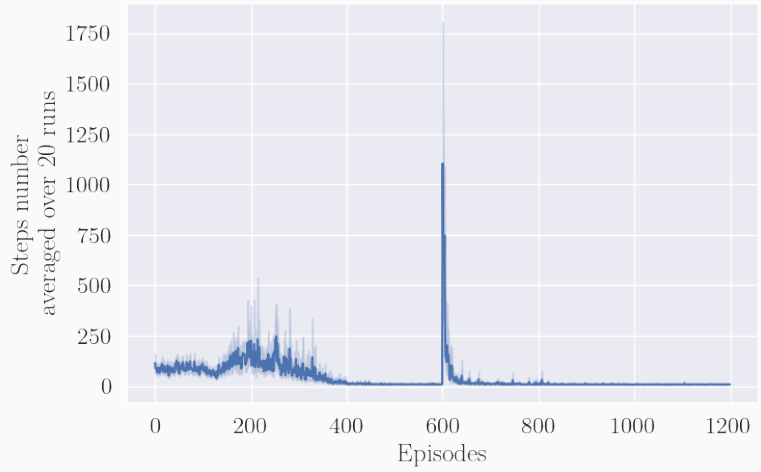


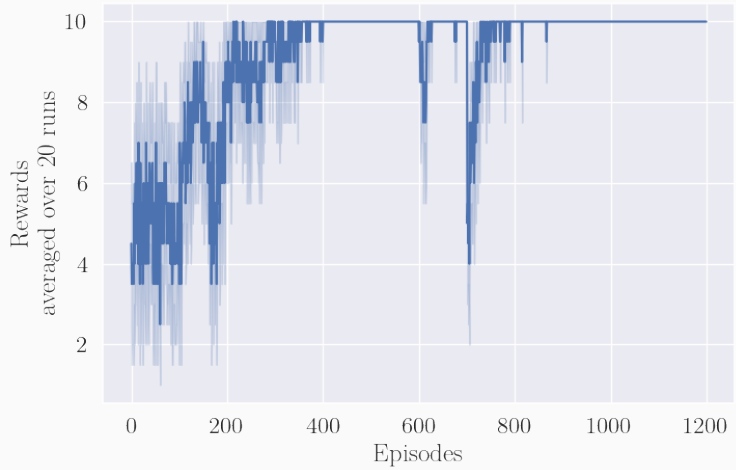


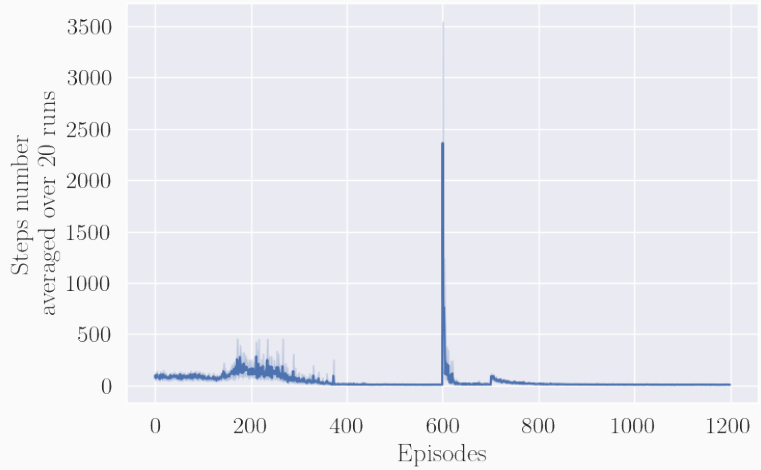


Training only in the lower triangle then switch to the upper triangle









Outline

1. Online Deep RL training
2. Generalization experiment
3. Discussion

Points of discussion

- Debrief from the meeting with Thomas
- Topics of discussion for future meetings?
  - How to compare neural data with simulation data?
  - Journal club (e.g. MINDS paper, etc.)
  - Any other topics to add?

Points of discussion

- Debrief from the meeting with Thomas
- Topics of discussion for future meetings?
  - How to compare neural data with simulation data?
  - Journal club (e.g. *MINDS* paper, etc.)
  - Any other topics to add?

Points of discussion

- Debrief from the meeting with Thomas
- Topics of discussion for future meetings?
  - How to compare neural data with simulation data?
  - Journal club (e.g. *MINDS* paper, etc.)
  - Any other topics to add?

Points of discussion

- Debrief from the meeting with Thomas
- Topics of discussion for future meetings?
  - How to compare neural data with simulation data?
  - Journal club (e.g. MINDS paper, etc.)
  - Any other topics to add?

Points of discussion

- Debrief from the meeting with Thomas
- Topics of discussion for future meetings?
  - How to compare neural data with simulation data?
  - Journal club (e.g. MINDS paper, etc.)
  - Any other topics to add?

Thanks!