$$J(\theta) = \mathop{E}_{y \sim p_\theta(\cdot|x)}\left[r(x,y) \right]$$

$$= \sum_y r(x,y)\, p_\theta(y|x)$$

$$\frac{dJ}{d\theta} = \sum_y r(x,y) \frac{d}{d\theta} p_\theta(y|x)$$

↳ Q: can we just sample a few y and compute the average of $r(x,y) \frac{d}{d\theta} p_\theta(y|x)$?

↳ A: no, because

$$\mathop{E}_{y \sim p_\theta(\cdot|x)}\left[r(x,y) \frac{d}{d\theta} p_\theta(y|x) \right]$$

$$= \sum_y \underbrace{p_\theta(y|x)}_{\text{from sampling!}}\, r(x,y) \frac{d}{d\theta} p_\theta(y|x)$$

dividing by this now yields log trick!

So from last class:

$$A(x,y) = r(x,y) - b(x)$$

baseline: 1. mean batch stats
2. group stats
3. trained value fn or critic
↳ RLHF

Value fn:

↳ takes prompt x and partial response $y_{<i}$, predicts $r(x,y)$

↳ just like reward model, scalar head trained on top of LM, L2 loss

↳ unlike RM, Value fn trained alongside policy model!

$$J(\theta) = \mathop{E}_{x,y \sim P_\theta}\left[A(x,y) - \beta D_{KL}(P_\theta \| P_{SFT})\right]$$

$$\log P_\theta(y|x) - \log P_{SFT}(y|x)$$

reverse KL penalty

$$= \sum_{i=1}^{n} \log P_\theta(y_i|x) - \log P_{SFT}(y_i|x)$$

we're not done yet!

↳ want to make small, stable changes when updating P_θ

↳ policy ratio $\rho_i = \dfrac{P_\theta(y_i|x)}{P_{\theta_{old}}(y_i|x)}$

$\rho_i = 1$

↳ $\theta = \theta_{old}$ for first step on batch, different after subsequent steps,

↳ how much more/less likely is this token under the new policy vs. the policy that generated it?

↳ we can weight our advantage by ρ_i, and control update by clipping ρ_i

Let's denote

$$A^{KL}_{(x,y)} = A(x-y) - \beta D_{KL}(P_\theta \| P_{SFT})$$

then

$$J(\theta) = E\left[\sum_{i=1}^{n} \min\left(\rho_i A_t^{KL}(x,y),\ \text{clip}(\rho_i, 1-\varepsilon, 1+\varepsilon) A_t^{KL}(x,y)\right)\right]$$

↑ clipping hyperparam = 0.1, 0.2

intuition:

↳ if $A_t^{KL} > 0$, we like this token and want to increase its prob

↳ max increase of $(1+\varepsilon)A_t^{KL}$

↳ if $A_t^{KL} < 0$, we don't like the token

↳ max decrease $(1-\varepsilon)A_t^{KL}$

↳ no gradient thru clipped term!

example:

$\varepsilon = 0.2$, so we clip $(\rho_i, 0.8, 1.2)$

$A_i = 2.0$, $\rho_i = 1.3$

↳ unclipped = 2.6

clipped = 2.4 ⇒ choose min; no gradients; token already more likely under new θ

$A_i = 2.0$, $\rho_i = 1.1$

↳ unclipped = 2.2, clipped = 2.4

↳ normal gradient, increase prob of token

$A_i = -2.0$, $\rho_i = 0.6$

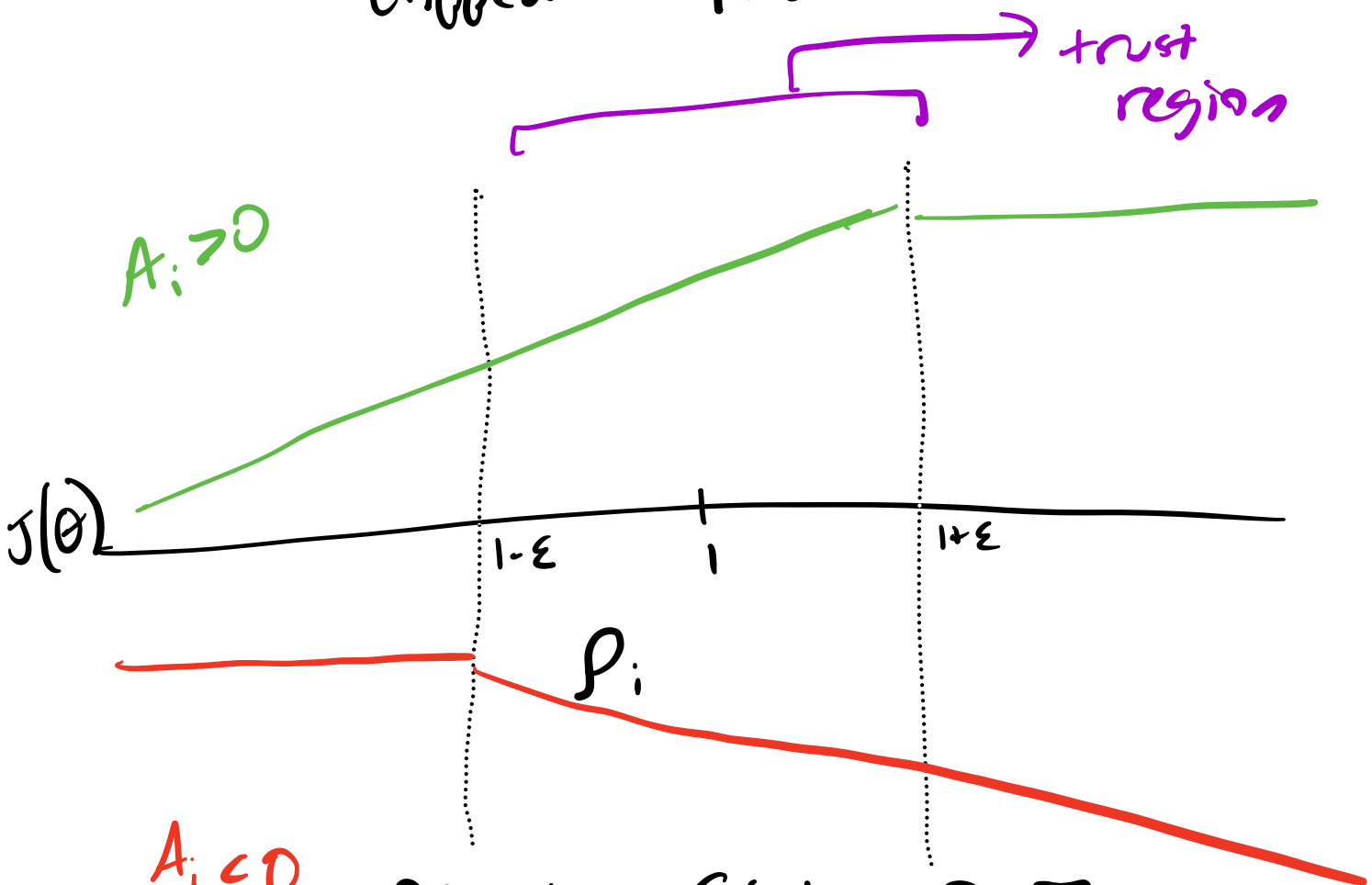
↳ unclipped = -1.2

clipped = -1.6 ⇒ choose min, no gradient, token already less likely under new θ

$A_i = -2.0$, $\rho_i = 0.9$

↳ unclipped = -1.8 → normal grad, decrease prob of token

clipped = -1.6



PPO algo: Schulman et al 2017

GRPO : group-relative PPO

↳ baseline comes from group statistics

↳ for each prompt x, sample a group of responses $y_1, y_2, \ldots, y_G$

↳ compute mean reward μ_x and std dev of reward σ_x over group

↳ for any specific y_i in group,

$$A(x, y_i) = \frac{r(x, y_i) - \mu_x}{\sigma_x}$$

↳ z-score

Deepseek R1:

- GRPO
- remove reward model, instead use simple verifiable tasks
- e.g. Math problems
  - prompt x: $37 + 53 = ?$
  - response y_1: 32 $\Rightarrow$ reward 0
    response y_2: 47 $\Rightarrow$ reward 0
    response y_3: 90 $\Rightarrow$ reward 1
- incentivize thinking / reasoning before answering