Let π_θ denote policy with parameter θ. The theoretical TRPO update is:

$$\theta_{k+1} = \text{argmax}_\theta L(\theta_k, \theta) \tag{1}$$
$$\text{s.t } \bar{D}_{KL}(\theta||\theta_k) \leq \delta \tag{2}$$

$L(\theta_k, \theta)$ is the *surrgate advantage*, a measure of how policy θ performs relative to the old policy θ_k on the trajectories sampled from θ_k.

$$L(\theta_k, \theta) = \mathbb{E}_{s,a\sim\theta_k}(\frac{\pi_\theta(a|s)}{\pi_{\theta_k}(a|s)} A^{\pi_{\theta_k}}(s,a)) \tag{3}$$

$L(\theta_k, \theta)$ is the importance sampling estimate of advantage of π_θ using trajectory generated from π_{θ_k}
$\bar{D}_{KL}(\theta||\theta_k)$ is the average KL-divergence between policies across states visited by the old policy

$$\bar{D}_{KL}(\theta||\theta_k) = \mathbb{E}_{s\sim\pi_k}[D_{KL}(\pi(\cdot|s)||\pi_{\theta_k}(\cdot|s))] \tag{4}$$

In other policy gradient based methods like REINFORCE or A2C, we estimate the policy gradient through the log of policy gain

$$L(\theta_k) = \mathbb{E}_{s,a\sim\theta_k}(\log \pi_{\theta_k}(a|s)A^k(s,a)) \tag{5}$$

The importance sampling interpretation of the gain Ł(θ_k, θ) and the log loss should produce the sample policy gradient

$$\frac{\partial L(\theta_k,\theta)}{\partial\theta} = \mathbb{E}(\frac{1}{\pi_\theta}\frac{\partial\pi_\theta(a|s)}{\partial\theta}A)|_{\theta=\theta_k} = \mathbb{E}(\frac{\partial}{\partial\theta}\log\pi_\theta(a|s)A)|_{\theta=\theta_k} = \frac{\partial}{\partial\theta}L(\theta) \tag{6}$$

One reason we wanted the importance sampling interpretation of the policy gain is that it can be used to for algorithms that are *slightly* off-policy and hence, improve the sample effciency. (see PPO)
Both $L(\theta_k, \theta)$ and $\bar{D}_{KL}(\theta||\theta_k)$ are not easy to estimate. So we approximate these quantities in TRPO.

$$L(\theta_k, \theta) \simeq g(\theta - \theta_k) \tag{7}$$
$$D_{KL}(\theta||\theta_k) \simeq \frac{1}{2}(\theta-\theta_k)H(\theta-\theta_k) \tag{8}$$

i.e. we use the first order approximation of $L(\theta_k, \theta)$ and second order approximation of $D_{KL}(\theta_k, \theta)$ (the first order term of D_{KL} vanishes because D_{KL} achieves minimum at $\theta = \theta_k$).
I don't know what justifies the use of first order approximation of $L(\theta_k, \theta)$
The approximate optimization problem becomes

$$\theta_{k+1} = \text{argmax}_\theta g(\theta - \theta_k) \tag{9}$$

$$\text{s.t.} \frac{1}{2}(\theta - \theta_k) H(\theta - \theta_k) \leq \delta \tag{10}$$

The solution to the above optimization problem can be solved analytically (via Lagrange method):

$$\theta_{k+1} = \theta_k + \sqrt{\frac{2\delta}{ng \cdot H \cdot ng}} ng \tag{11}$$

where ng is the *natural policy gradient* $H^{-1}g$.
This can be simplified into

$$\theta_{k+1} = \theta_k + \sqrt{\frac{2\delta}{g^T H^{-1} g}} H^{-1} g \tag{12}$$

Due the approximation of optimization objective and sampling estimate of KL, we do not know if the update satisfies the KL constraint or if the policy gain improves. TRPO add those safeguards by doing a backtracking line search.

0.1 Why natural gradient is interesting

The more interesting part of the update step is the direction of the update

$$H^{-1} g \tag{13}$$

it is called *natural gradient* because it is the actual gradient if we view π_θ as a point on the policy manifold rather than a point in $\mathbb{R}^n$.
Let Π denote the policy manifold, it is diffeomorphism (it means if you zoom in, the map looks like invertible linear map) to $\mathbb{R}^n$ by the natural maps

$$\pi_\theta \leftrightarrow \theta$$

So the only way to make Π a more interesting manifold is to equip it with a non-Euclidean metric.
(Amari 1985, Rao 1945) Given a family of parametric probability distributions $p(x, w)$ over X ($x \in X$ and w is the parameter), there is unique Riemannian metric on $p(x, w)$

$$g_{ij}(w) = \mathbb{E}[\frac{\partial \log p(x, w)}{\partial w_i} \frac{\log p(x, w)}{\partial w_j}] \tag{14}$$

Moreover, this is only invariant metric to be given to $p(x, w)$.

This means if we have a change of coordinate $y = f(w)$ on $\mathbb{R}^n$, then the length of the vector in y measured with g_y is the same as the (same) vector in w measured in g_w.
[**HW**] verify it.
Let (M, g) be a Riemannian manifold and let $f : M \to \mathbb{R}$ be a global function, how to compute ∇f?
Let $df : TM \to TR$ be the induced map on tangent space, then ∇f is characterized as the following: for any $Y \in TM$,

$$df(Y) = \langle \nabla f, Y \rangle \tag{15}$$

Suppose M has global coordinate $x_1, \cdots, x_n$ (like Π), then

$$df = \frac{df}{dx_1} dx_1 + \cdots + \frac{df}{dx_n} \tag{16}$$

I am expressing df in terms of the basis elements of $TM*$, dx_i maps the unit tangent vector along x_i direction (in Euclidean sense) to 1 and tangent vectors along other axes to 0.
So when you compute the "normal" gradient via backprop, you are in fact computing the differential.
By abusing notation, write $df = \langle \frac{df}{dx_1}, \cdots, \frac{df}{dx_n} \rangle$ (vector form). Write $g = [g_{ij}]$, the metric on M in matrix form. Then

$$\nabla f = g^{-1} df \tag{17}$$

The right-hand-side is the usual matrix vector product.
Conclusion: natural gradient is the real gradient.

0.2 How is it related to the KL business

H in the optmization objective is the Hessian of the

$$\frac{1}{N} \sum_{1}^{N} \frac{\partial^2}{\partial\theta_i \partial\theta_j} KL(\pi_\theta(\cdot|s_n) || \pi_\theta(\cdot|s_n)) \tag{18}$$

computed analytically. It is the same thing as integrating

$$\frac{1}{N} \sum_{n=1}^{N} \frac{\partial}{\partial\theta_i} \log \pi_\theta(a_n|s_n) \frac{\partial}{\partial\theta_j} \log \pi_\theta(a_n|s_n) \tag{19}$$

(the Fisher information matrix over (a_n, s_n)) over the action space.
Pseudocode:

Algorithm 1: Trust Region Policy Optimization

1: Input: initial policy parameters θ_0, initial value function parameters ϕ_0;
2: Hyperparameters: KL-divergence limit δ, backtracking coefficient α, maximum number of backtracking steps K

for $k = 0, 1, 2, ...,$ **do**

collect a set of trajecotries $D_k = \{\tau_i\}$ by running policy π_k in the environment

Compute rewards-to-go $\hat{R}_t$.

Compute advantage estimates, $\hat{A}_t$ based on the current value function V_{ϕ_k}

Estimate policy gradient as

$$\hat{g}_k = \frac{1}{|D_k|T} \sum_{\tau \in D_k} \sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(a_t|s_t)|_{\theta_k} \hat{A}_t$$

Use conjugate gradient algorithm to compute the natural policy gradient

$$\hat{x} \simeq \hat{H}_k^{-1} \hat{g}_k \tag{20}$$

where $\hat{H}_k$ is the Hessian of the sample averge of KL-divergence

Update the policy by backtracking line search with

$$\theta_{k+1} = \theta_k + \alpha^j \sqrt{\frac{2\delta}{\hat{x}_k^T \hat{H}_k \hat{x}_k}} \hat{x}_k$$

where $j \in \{0, 1, 3, \cdots, K\}$ is the smallest value which improves the sample loss and satisfies the KL-divergence constraint.

Fit value function by regression on mean-squared error:

$$\phi_{k+1} = \text{argmin}_\phi \frac{1}{|D_k|T} \sum_{\tau \in D_k} \sum_{t=0}^{T} (V_\phi(s_t) - \hat{R}_t)^2 \tag{21}$$

end for

A few assumptions:

1. The policy π does not change the environment, i.e. it has a well-defined stationary distribution ρ^π. It is more technically called *ergodic*. This is why you should not believe RL, implemented naively, can make you rich in financial market.

Notes from (Kakade 2002, A Natural Policy Gradient) Interesting things I have not thought about

How to think about the steepest ascend direction? Let $\eta(\theta)$ be the average reward of the policy π_θ. What is the steepest direction $d\theta$. It is the direction that maximizes $\eta(\theta + d\theta)$ under the constraint that the length $|d\theta|^2$ is held small constant. The length is defined with respect to the *metric* on the policy manifold.

Theoretical justification of actor-critic methods

Suppose $Q^\pi(s, a)$ is approximated by some *compatible* function approximator $f^\pi(s, a; \omega)$. For vectors $\theta, \omega \in \mathbb{R}^m$, we define

$$\phi(s, a)^\pi = \nabla \log \pi(a; s, \theta), f^\pi(s, a; \omega) = \omega^T \phi(s, a)^\pi \tag{22}$$

Suppose $\tilde{\omega}$ minimizes the square error

$$\epsilon(\omega, \pi) = \sum_{s,a} \rho^{\pi}(s)\pi(a; s, \theta)(f^{\pi}(s, a; \omega) - Q^{\pi}(s, a))^2 \tag{23}$$

The function app $f^{\pi}(s, a; \omega)$ is *compatible* with the policy in the sense that it can be used in lieu of $Q^{\pi}(s, a)$ to calculate the policy gradient. The result will be exact.
Simple proof, just differentiate $\epsilon(\omega, \pi)$.

Theorem 1 *Let $\tilde{\omega}$ minimizes the squared error $\sigma(\omega, \theta)$. Then, $\tilde{\omega}$ is the natural policy gradient.*

If the function approximator we use looks like $\omega\phi^{\pi}(s, a)$ then this theorem shows why natural gradient is a good choice. However, why should we assume function approximator of Q looks like $\omega\phi^{\pi}(s, a)$? Why the log derivative has anything with the state action value?
Why exponential family of policies? Because the math can work, if you move a point along the tangent direction, the point would still be on the manifold (the policy manifold).