Subspace Inference for Bayesian Deep Learning

Pavel Izmailov[*1] **Wesley Maddox**[*1] **Polina Kirichenko**[*1] **Timur Garipov**[*2,3] **Dmitry Vetrov**[2,3] **Andrew Gordon Wilson**[1]

[1]Cornell University, [2]Higher School of Economics, [3]Samsung-HSE Laboratory, [4]Samsung AI Center in Moscow

Outline

Bayesian inference provides full predictive distributions and well calibrated uncertainty estimates, but is challenging to perform for modern deep neural networks with a high dimensional weight space.

- **What?** We design a low-dimensional subspace $\mathcal{S}$ of the weight space and approximate the posterior over the parameters in this subspace.
- **Why?** Standard Bayesian inference procedures scale poorly with dimensionality.
- **How?** We use information contained in SGD trajectory to construct the subspace; we use Variational Inference or Elliptical Slice Sampling for approximate inference in the subspace.

Result: Even when using exceptionally low dimensional subspaces, Bayesian inference is possible on large modern neural networks with minimal computational overhead.

Inference within a Subspace

Assume a set of $K+1$ vectors $\{v_1, v_2, \dots, v_K, \hat{w}\}$ in the full weight space, $\mathbb{R}^p$; define subspace:

$$\mathcal{S} = \{w | w = \hat{w} + t_1 v_1 + \dots t_K v_K\} = \{w | w = \hat{w} + Pt\},$$

with $\hat{w} \in \mathbb{R}^p$, $P = (v_1^T, \dots, v_K^T) \in \mathbb{R}^{p \times K}$, and $t = (t_1, \dots, t_K)^T \in \mathbb{R}^K$.

New likelihood is a function of t:

$$p(\mathcal{D}|t) = p_{\mathcal{M}}(\mathcal{D}|w = \hat{w} + Pt).$$

Bayesian model averaging:

$$p(\mathcal{D}^*|\mathcal{D}) = \sum_i p_{\mathcal{M}}(\mathcal{D}^*|\tilde{w} = \hat{w} + P\tilde{t}_i), \quad \tilde{t}_i \sim q(t|\mathcal{D}),$$

where $q(t|\mathcal{D})$ is approximate posterior over t, represented by MCMC samples or a deterministic (variational) approach.

Posterior Tempering. #parameters << #data points in the subspace model, hence posterior over t is extremely concentrated. Instead, we utilize the *tempered* posterior:

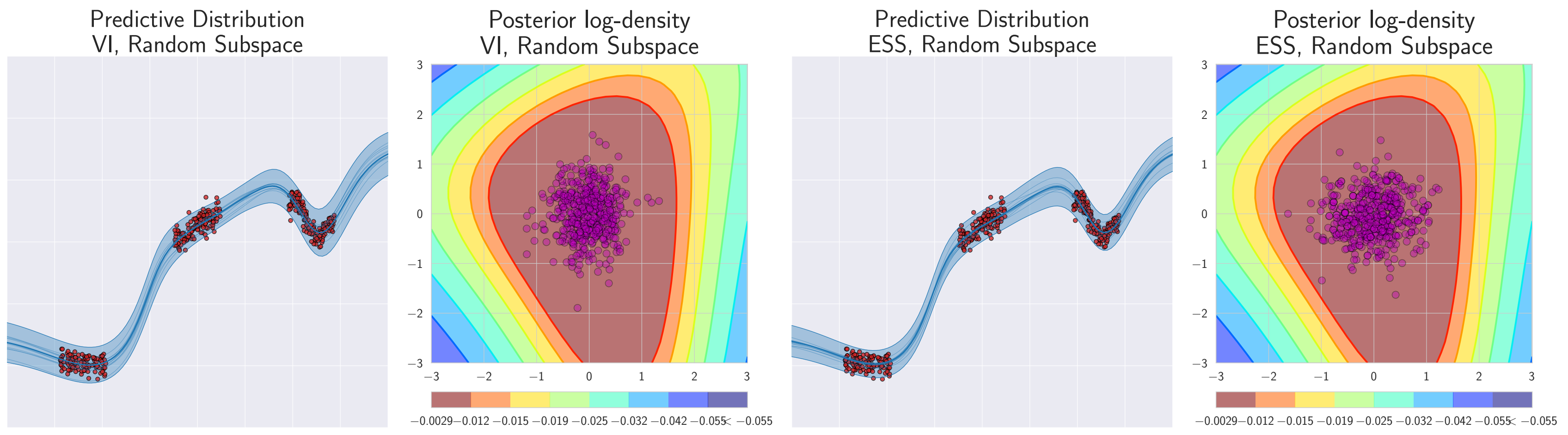
$$p_T(t|\mathcal{D}) \propto \underbrace{p(\mathcal{D}|t)}_{\text{likelihood}}{}^{1/T} \underbrace{p(t)}_{\text{prior}}$$

We choose T by cross validation.

Subspace Construction

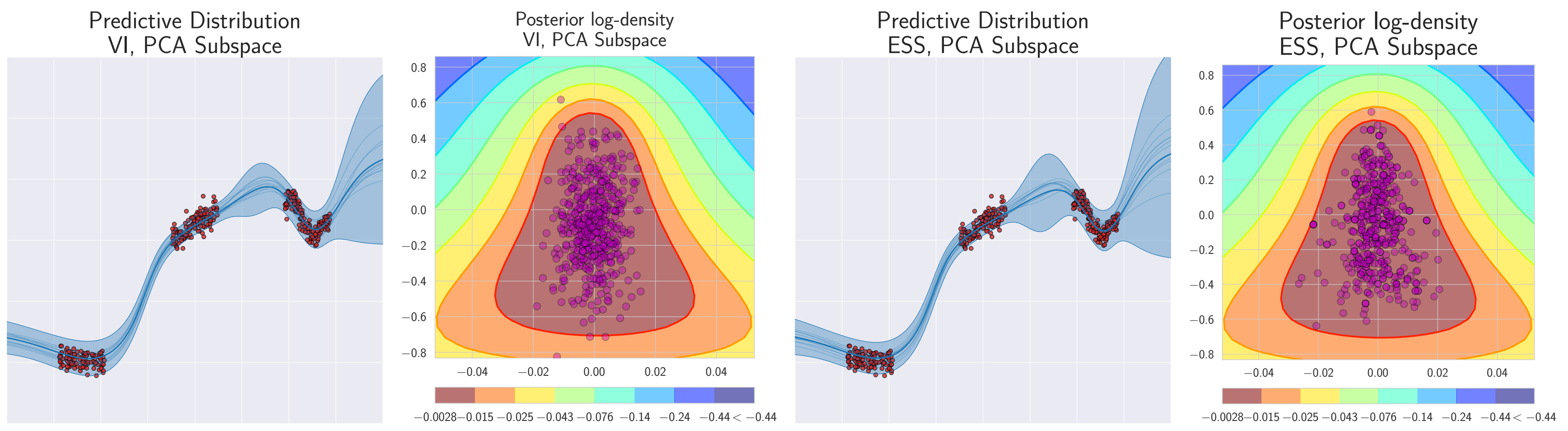
Intuitively we want the subspace $\mathcal{S}$ to (1) contain a diverse (producing meaningfully different predictions on test data) set of models and (2) be cheap to construct.

Random Subspaces



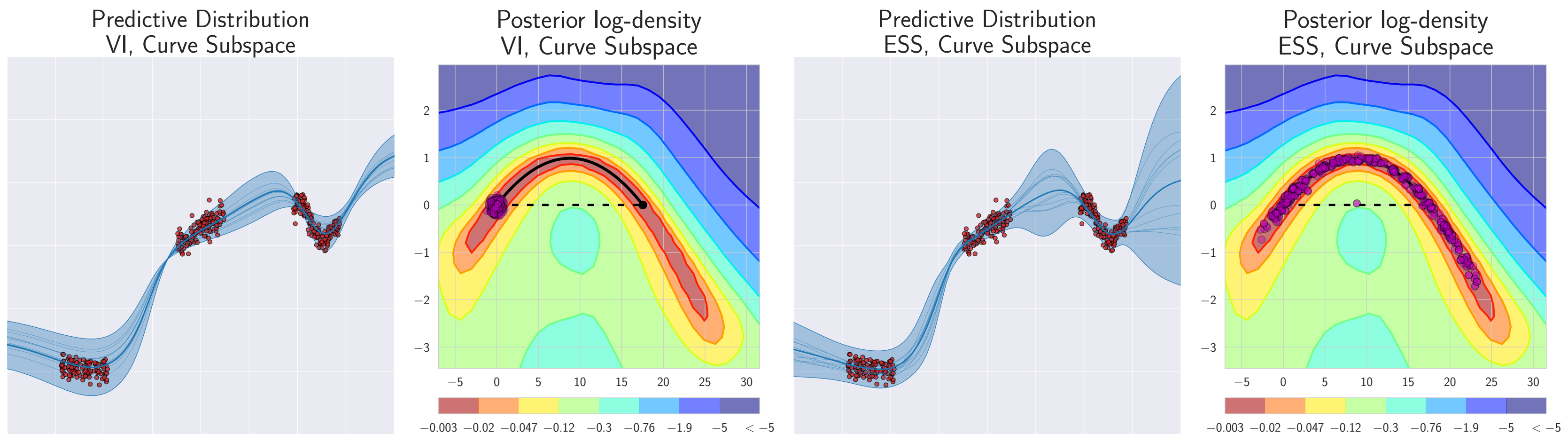
- Directions $v_1, \dots, v_K \sim \mathcal{N}(0, I_p)$
- Use a pre-trained solution as shift $\hat{w}$
- Prior $\mathcal{N}(0, \sigma^2 I_K)$

PCA of the SGD Trajectory



- Run SGD with high constant learning rate from a pre-trained solution and collect snapshots w_i of weights
- Use SWA solution as shift $\hat{w} = \frac{1}{T}\sum w_i$
- $\{v_1, v_2, \dots, v_k\}$ – first K PCA components of vectors $w_i - \hat{w}$
- Prior $\mathcal{N}(0, I)$

Curve Subspaces



- Garipov et al. 2018 proposed a method to find two-dimensional subspaces containing a path of low loss between weights of two independently trained neural networks
- Isotropic Gaussian prior $N(0, \sigma^2 I)$

Experiments

Table 1: Negative log-likelihood and Accuracy for PreResNet-164 for 10-dimensional random, 10-dimensional PCA, and 2-dimensional curve subspaces.

	Random	PCA	Curve
NLL	0.6858 ± 0.0052	0.6652 ± 0.004	0.6464 ± 0.01
Accuracy (%)	80.17 ± 0.03	80.54 ± 0.13	81.28 ± 0.26

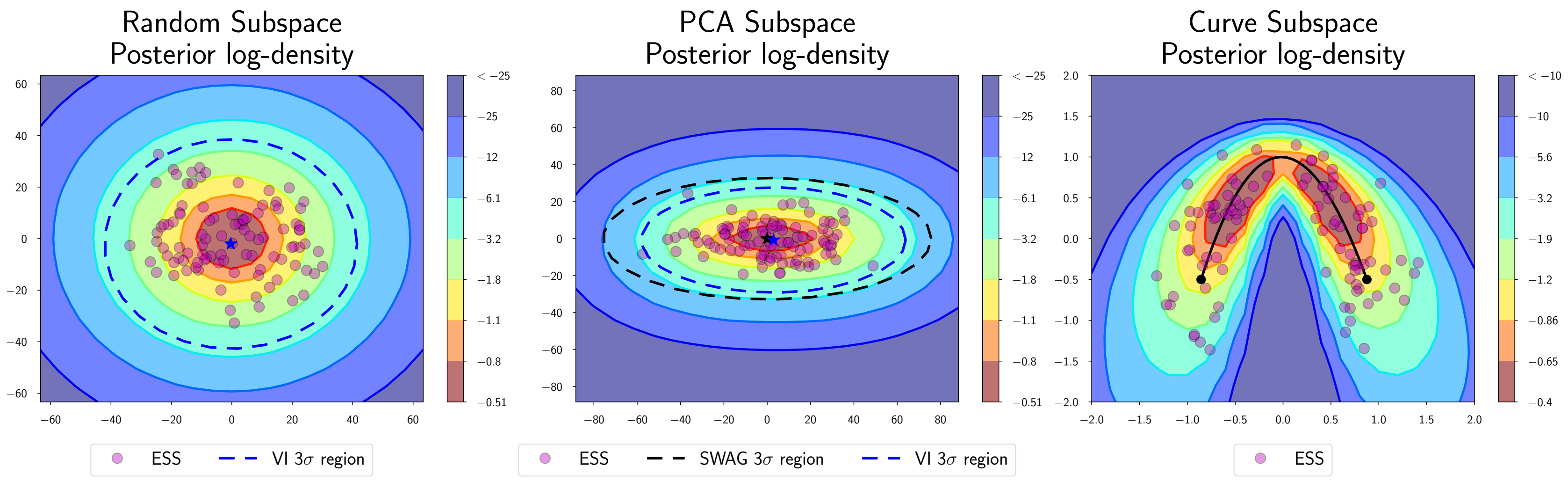


Figure 1: Posterior log-density surfaces, ESS samples, and VI approximate posterior distribution in **(a)** random, **(b)** PCA and **(c)** curve subspaces for PreResNet-164 on CIFAR-100.

- Curve Subspace > PCA Subspace > Random Subspace
- PCA subspace generally has the best run-time accuracy trade-off.
- Despite its simplicity, Subspace Inference in the PCA subspace is competitive with many popular alternatives: SWAG, MC-Dropout and Temperature Scaling on image classification and UCI regression data.

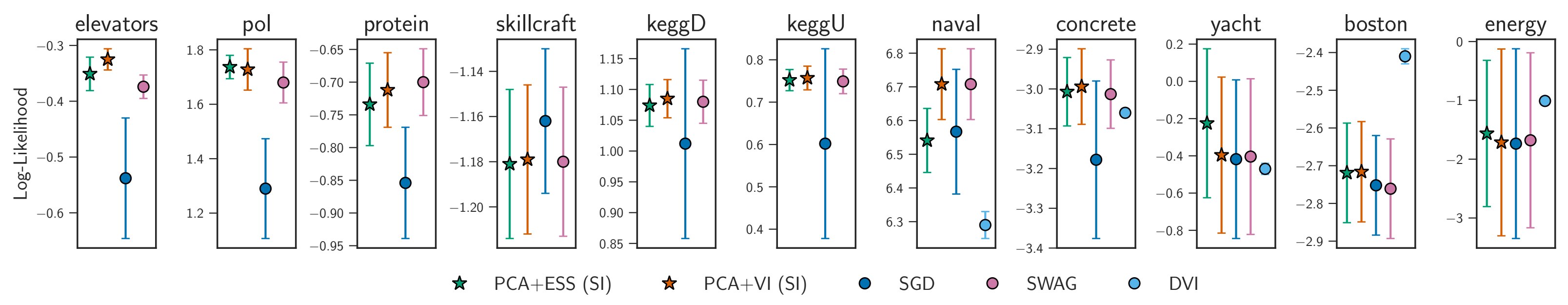


Figure 2: Test log-likelihoods for proposed methods on six UCI regression datasets.