A. Proof of Biases for CG and RFF

A.1. Proof of Theorem 1

Proof. To prove the bias of the CG log marginal likelihood terms, we rely on a connection between conjugate gradients, the Lanczos algorithm (Lanczos, 1950), and Gauss quadrature. In particular, we demonstrate that u_J and v_J – CG's estimates for $\mathbf{y}^\top \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1}\mathbf{y}$ and $\log|\widehat{\mathbf{K}}_{\mathbf{XX}}|$ terms – are equivalent to Gauss quadrature approximations of two Riemann-Stieljes integrals. We then use quadrature error analysis to prove the biases of these terms.

Expressing $\mathbf{y}^\top \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1}\mathbf{y}$ and $\mathbf{z}^\top(\log \widehat{\mathbf{K}}_{\mathbf{XX}})\mathbf{z}$ as Riemann-Stieljes integrals. First, we note that $\mathbf{y}^\top \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1}\mathbf{y}$ and $\mathbf{z}^\top(\log \widehat{\mathbf{K}}_{\mathbf{XX}})\mathbf{z}$ (our stochastic trace estimate of $\log|\widehat{\mathbf{K}}_{\mathbf{XX}}|$) can both be expressed by the quadratic form $\mathbf{w}^\top f(\mathbf{A})\mathbf{w}$, where $f(\mathbf{A})$ denotes a matrix function of the matrix $\mathbf{A}$, and $\mathbf{w}$ is a vector. Letting $\mathbf{A} = \mathbf{P}\mathbf{\Lambda}\mathbf{P}^\top$ be the eigendecomposition of $\mathbf{A}$, we can write this quadratic form as

$$\mathbf{w}^\top f(\mathbf{A})\mathbf{w} = \mathbf{w}^\top \mathbf{P} f(\mathbf{\Lambda})\mathbf{P}^\top \mathbf{w} = \|\mathbf{w}\|^2 \sum_{i=1}^{N} f(\lambda_i)\mu_i^2,$$

where $\lambda_{\min} \le \lambda_i \le \lambda_{\max}$ are the diagonal elements of $\mathbf{\Lambda}$ (i.e. the eigenvalues) – ordered from smallest to largest, and μ_i are the components of $\mathbf{P}^\top\mathbf{w}/\|\mathbf{w}\|$. The summation in the above equation can be expressed as a Riemann-Stieltjes integral:

$$\begin{aligned} I[f] &:= \|\mathbf{w}\|^2 \sum_{i=1}^{N} f(\lambda_i)\mu_i^2 \\ &= \|\mathbf{w}\|^2 \int_{\lambda_{\min}}^{\lambda_{\max}} f(t) d\mu_{\mathbf{A}}(t), \end{aligned} \tag{S1}$$

where the measure $\mu_{\mathbf{A}}(t)$ is a piecewise constant function

$$\mu_{\mathbf{A}}(t) = \begin{cases} 0, & \text{if } t < \lambda_{\min}, \\ \sum_{j=1}^{i} \mu_j^2, & \text{if } \lambda_i \le t < \lambda_{i-1} \\ \sum_{j=1}^{N} \mu_j^2, & \text{if } \lambda_{\max} \le t. \end{cases} \tag{S2}$$

See (Golub & Meurant, 2009) for more details.

The Lanczos algorithm approximates these integrals with Gauss quadrature. The Lanczos algorithm (Lanczos, 1950), which is briefly described in Sec. 3.1, iteratively expresses a symmetric matrix $\mathbf{A}$ via the partial tridiagonalization $\mathbf{T}_{\mathbf{w}}^{(J)} = \mathbf{Q}_{\mathbf{w}}^{(J)\top}\mathbf{A}\mathbf{Q}_{\mathbf{w}}^{(J)}$. $\mathbf{Q}_{\mathbf{w}}^{(J)} \in \mathbb{R}^{N\times J}$ is a orthonormal matrix with $\mathbf{w}/\|\mathbf{w}\|$ as its first column, and $\mathbf{T}_{\mathbf{w}}^{(J)}$ is a $J \times J$ tridiagonal matrix. Briefly, the columns of $\mathbf{Q}^{(J)}$ matrices are computed by performing Gram-Schmidt orthogonalization on the Krylov subspace $[\mathbf{w}, \mathbf{A}\mathbf{w}, \mathbf{A}^2\mathbf{w}, \ldots, \mathbf{A}^{J-1}\mathbf{w}]$, and storing the orthogonalization constants in $\mathbf{T}_{\mathbf{w}}^{(J)}$.

The Lanczos algorithm is commonly used to estimate quadratic forms:

$$\begin{aligned} \mathbf{w}^\top f(\mathbf{A})\mathbf{w} &\approx \mathbf{w}^\top \mathbf{Q}^{(J)})_{\mathbf{w}} f\left(\mathbf{T}_{\mathbf{w}}^{(J)}\right) \mathbf{Q}^{(J)\top})_{\mathbf{w}}\mathbf{w} \\ &= \|\mathbf{w}\|^2 \mathbf{e}_1^\top f\left(\mathbf{T}_{\mathbf{w}}^{(J)}\right)\mathbf{e}_1, \end{aligned} \tag{S3}$$

where $\mathbf{e}_1$ is the first unit vector. Note that Eq. (S3) holds because the columns of $\mathbf{Q}_{\mathbf{w}}^{(J)}$ are orthonormal.

There is a well-established connection between Eq. (S3) and numeric quadrature (e.g. Golub & Meurant, 2009). More specifically, Eq. (S3) is exactly equivalent to a J-term Gauss quadrature rule applied to the Riemann-Stieltjes integral in Eq. (S1). We can thus analyze Lanczos estimates of $\mathbf{w}^\top f(\mathbf{A})\mathbf{w}$ using standard Gauss quadrature error analysis.

Equivalence between CG and the Lanczos algorithm. We will now show an equivalence between our estimates $u_J \approx \mathbf{y}^\top \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1}\mathbf{y}$ and $v_J \approx \mathbf{z}^\top(\log \widehat{\mathbf{K}}_{\mathbf{XX}})\mathbf{z}$ and Lanczos algorithm approximations. Note that we have already established $v_J \approx \mathbf{z}^\top(\log \widehat{\mathbf{K}}_{\mathbf{XX}})\mathbf{z}$ as a Lanczos algorithm approximation in Eq. (8):

$$\mathbf{z}^\top = \|\mathbf{z}\|^2 \mathbf{e}_1^\top (\log \mathbf{T}_{\mathbf{z}}^{(J)})\mathbf{e}_1.$$

For $u_J \approx \mathbf{y}^\top \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1}\mathbf{y}$, we exploit a connection between CG and Lanczos (Golub & Van Loan, 2012, Ch. 11.3):

$$\sum_{j=1}^{J} \gamma_j \mathbf{d}_j = \|\mathbf{y}\| \mathbf{Q}_{\mathbf{y}}^{(J)} (\mathbf{T}_{\mathbf{y}}^{(J)})^{-1}\mathbf{e}_1, \tag{S4}$$

where $\sum_{j=1}^{J} \gamma_j \mathbf{d}_j$ (see Eq. 4) is the J^{th} CG approximation to $\widehat{\mathbf{K}}_{\mathbf{XX}}^{-1}\mathbf{y}$. Multiplying Eq. (S4) by $\mathbf{y}^\top$, we have

$$\begin{aligned} u_J &= \mathbf{y}^\top \left(\sum_{j=1}^{J} \gamma_j \mathbf{d}_j\right) \\ &= \|\mathbf{y}\| \mathbf{y}^\top \mathbf{Q}_{\mathbf{y}}^{(J)} (\mathbf{T}_{\mathbf{y}}^{(J)})^{-1}\mathbf{e}_1 \\ &= \|\mathbf{y}\|^2 \mathbf{e}_1^\top (\mathbf{T}_{\mathbf{y}}^{(J)})^{-1}\mathbf{e}. \end{aligned}$$

Therefore, the CG approximations of $\mathbf{y}^\top \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1}\mathbf{y}$ and $\mathbf{z}^\top(\log \widehat{\mathbf{K}}_{\mathbf{XX}})\mathbf{z}$ are both Lanczos approximations. Putting all the pieces together, we have just shown that u_J and v_J are equivalent to J-term Gauss quadrature rules applied to the following integrals:

$$\mathbf{y}^\top \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1}\mathbf{y} = \|\mathbf{y}\|^2 \int_{\lambda_{\min}}^{\lambda_{\max}} t^{-1}\, \mathrm{d}\mu_{\widehat{\mathbf{K}}_{\mathbf{XX}}}(t), \tag{S5}$$

$$\mathbf{z}^\top \left(\log \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1}\right)\mathbf{z} = \|\mathbf{z}\|^2 \int_{\lambda_{\min}}^{\lambda_{\max}} \log(t)\, \mathrm{d}\mu_{\widehat{\mathbf{K}}_{\mathbf{XX}}}(t), \tag{S6}$$

where the measure $\mu_{\widehat{\mathbf{K}}_{\mathbf{XX}}}(t)$ is defined by Eq. (S2).

Applying Gauss quadrature error analysis to u_J and v_J. To analyze the bias of teh CG estimates, we make use of standard error Guass quadrature error analysis. If $L_G^{(J)}[f]$ is the J-term Gaussian quadrature approximation of the Riemann-Stieltjes integral $I[f]$ (see Eq. S1), the error can be exactly expressed as:

$$I[f] - L_G^{(J)}[f] = (\gamma_1 \cdots \gamma_{J-1})^2 \frac{f^{(2J)}(\eta)}{(2J)}, \tag{S7}$$

for some $\eta \in [\lambda_{\min}, \lambda_{\max}]$, where $f^{(2J)}$ is the $2J^{\text{th}}$ derivative of f and $\{\gamma_i\}$ are some quantities that depend on the spectrum of the matrix $\widehat{\mathbf{K}}_{\mathbf{XX}}$ (Golub & Meurant, 2009, Ch. 6).

Turning back to u_J, the corresponding function $f(t) = t^{-1}$ has even derivatives of the form $f^{(2J)} = (2J)!\, t^{-(2J+1)} > 0, \forall J, \forall t \in (\lambda_{\min}, \lambda_{\max})$. Replacing $I[f]$ with the integral defined by Eq. (S5), we have that $\mathbf{y}^\top \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1}\mathbf{y} - u_J \geq 0$, which proves that CG underestimates $\mathbf{y}^\top \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1}\mathbf{y}$. Similarly for v_J, the corresponding $f(t) = \log t$ has even derivatives of the form $f^{(2J)} = -(2J-1)!\, t^{-2J} < 0, \forall J, \forall t \in (\lambda_{\min}, \lambda_{\max})$. Replacing $I[f]$ with the integral defined by Eq. (S6), we have that $\mathbf{z}^\top (\log \widehat{\mathbf{K}}_{\mathbf{XX}})\mathbf{z} - v_J \leq 0$.

The convergence rates of u_J and v_J follow from Eq. 4.4 of Ubaru et al. (2017). Let $\rho = (\sqrt{\kappa}+1)/(\sqrt{\kappa}-1)$, where κ is the condition number of $\widehat{\mathbf{K}}_{\mathbf{XX}}$ (i.e. the ratio of its maximum and minimum singular values). Then

$$\|\mathbf{y}^\top \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1}\mathbf{y} - u_J\| \leq C\rho^{-2J},$$
$$\|\mathbf{z}^\top (\log \widehat{\mathbf{K}}_{\mathbf{XX}})\mathbf{z} - v_J\| \leq C\rho^{-2J},$$

where C is a constant that depends on the extremal eigenvalues of $\widehat{\mathbf{K}}_{\mathbf{XX}}$. □

A.2. Proof of Theorem 2

Proof. The inequalities are a result of applying Jensen's inequality to the inverse of a positive definite matrix (which is a convex function) and the log determinant of a positive definite matrix (which is a concave function). For example, for the $\mathbf{y}^\top \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1}\mathbf{y}$ term we have that

$$\begin{aligned}
\mathop{\mathbb{E}}_{\mathbb{P}(\boldsymbol{\omega})}\left[\mathbf{y}^\top \widetilde{\mathbf{K}}_J^{-1}\mathbf{y}\right] &= \mathbf{y}^\top \mathop{\mathbb{E}}_{\mathbb{P}(\boldsymbol{\omega})}\left[\widetilde{\mathbf{K}}_J^{-1}\right]\mathbf{y} \\
&\geq \mathbf{y}^\top \left(\mathop{\mathbb{E}}_{\mathbb{P}(\boldsymbol{\omega})}\left[\widetilde{\mathbf{K}}_J\right]\right)^{-1}\mathbf{y} \\
&= \mathbf{y}^\top \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1}\mathbf{y}
\end{aligned}$$

A similar procedure, but in the opposite direction applies to the $\log|\widehat{\mathbf{K}}_{\mathbf{XX}}|$.

To estimate the rate of decay of the biases, we rely on two keys ideas. The first idea is to translate the central moments of the kernel matrix being approximated by the random Fourier features into the central moments of these features. This strategy resembles the analysis in (Nowozin, 2018) which uses (Angelova, 2012, Theorem 1). The second idea is to use an approximation to two matrix functions. For the inverse function we use a Neumann series and for the logarithm we use a Taylor series. These series require that the eigenvalues of the approximation residual $(\widehat{\mathbf{K}}_{\mathbf{XX}} - \widetilde{K}_J)$ are less than 1. We will make this assumption as we know that for a large enough J, the kernel approximation will be close to $\widehat{\mathbf{K}}_{\mathbf{XX}}$. Below is a formal argument.

For a fixed $\boldsymbol{\omega}$ we can write

$$\begin{aligned}
\mathbf{y}^\top \widetilde{\mathbf{K}}_J^{-1}\mathbf{y} &= \mathbf{y}^\top \left(\widehat{\mathbf{K}}_{\mathbf{XX}} - \widehat{\mathbf{K}}_{\mathbf{XX}} + \widetilde{\mathbf{K}}_J\right)^{-1}\mathbf{y} \\
&= \mathbf{y}^\top \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\left(\mathbf{I} - \mathbf{I} + \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\widetilde{\mathbf{K}}_J\widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\right)^{-1}\widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\mathbf{y} \\
&= \mathbf{y}^\top \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\left(\mathbf{I} - \left(\mathbf{I} - \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\widetilde{\mathbf{K}}_J\widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\right)\right)^{-1}\widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\mathbf{y}
\end{aligned} \tag{S8}$$

this last form allow us to use a Neumann series to expand the inner inverse matrix. Hence, we have that

$$\begin{aligned}
&\left(\mathbf{I} - \left(\mathbf{I} - \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\widetilde{\mathbf{K}}_J\widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\right)\right)^{-1} \\
&= \sum_{t=0}^{\infty}\left(\mathbf{I} - \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\widetilde{\mathbf{K}}_J\widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\right)^t
\end{aligned}$$

Combining this with Eq. (S8), we have that

$$\begin{aligned}
&\mathop{\mathbb{E}}_{\mathbb{P}(\boldsymbol{\omega})}\left[\mathbf{y}^\top \widetilde{\mathbf{K}}_J^{-1}\mathbf{y}\right] = \mathbf{y}^\top \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1}\mathbf{y} \\
&+ \sum_{t=2}^{\infty}\mathbf{y}^\top \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\mathop{\mathbb{E}}_{\mathbb{P}(\boldsymbol{\omega})}\left[\left(\mathbf{I} - \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\widetilde{\mathbf{K}}_J\widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\right)^t\right]\widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\mathbf{y}
\end{aligned} \tag{S9}$$

where the first term of the series ($t = 0$) is the data-fit term being approximated and the second term of the series cancels out ($t = 1$). Following the analysis in (Nowozin, 2018) we translate the central moments of the random variable $\widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\widetilde{\mathbf{K}}_J\widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}$ to the central moments of $\widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\left(\boldsymbol{\phi}(\boldsymbol{\omega})\boldsymbol{\phi}(\boldsymbol{\omega})^\top + \sigma^2\mathbf{I}\right)\widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}$ denoted as μ_i for $i \geq 2$. Since the term ($t = 2$) will dominate, we will only focus on it (as the others would be of a higher order in J, see (Angelova, 2012)). We have that

$$\mathop{\mathbb{E}}_{\mathbb{P}(\boldsymbol{\omega})}\left[\left(\mathbf{I} - \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\widetilde{\mathbf{K}}_J\widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\right)^2\right] = \frac{\mu_2}{J} \tag{S10}$$

Therefore, incorporating the previous result into Eq. (S9), allows us to conclude that

$$\mathop{\mathbb{E}}_{\mathbb{P}(\boldsymbol{\omega})}\left[\mathbf{y}^\top \widetilde{\mathbf{K}}_J^{-1}\mathbf{y}\right] - \mathbf{y}^\top \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1}\mathbf{y} = \mathcal{O}(1/J)$$

For the model complexity term $\log|\widehat{\mathbf{K}}_{\mathbf{XX}}|$ we follow a similar procedure as before. For a fixed $\boldsymbol{\omega}$

$$\begin{aligned}&\log\left|\widetilde{\mathbf{K}}_J\right| = \log\left|\widehat{\mathbf{K}}_{\mathbf{XX}} - \widehat{\mathbf{K}}_{\mathbf{XX}} + \widetilde{\mathbf{K}}_J\right| \\ &= \log\left|\widehat{\mathbf{K}}_{\mathbf{XX}}\right| - \log\left|\mathbf{I} - \left(\mathbf{I} - \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\widetilde{\mathbf{K}}_J\widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\right)\right|\end{aligned}$$

Focusing on the last term we have that

$$\begin{aligned}&\log\left|\mathbf{I} - \left(\mathbf{I} - \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\widetilde{\mathbf{K}}_J\widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\right)\right| \\ &= \operatorname{tr}\left(\log\left(\mathbf{I} - \left(\mathbf{I} - \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\widetilde{\mathbf{K}}_J\widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\right)\right)\right)\end{aligned}$$

We can rewrite the term in the R.H.S as follows

$$\begin{aligned}&\log\left(\mathbf{I} - \left(\mathbf{I} - \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\widetilde{\mathbf{K}}_J\widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\right)\right) \\ &= \sum_{t=1}^{\infty}\frac{1}{t}\left(\mathbf{I} - \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\widetilde{\mathbf{K}}_J\widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\right)^t\end{aligned}$$

Again, following the analysis of (Nowozin, 2018), we can express the central moments of the random variable $\widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\widetilde{\mathbf{K}}_J\widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}$ to the central moments of $\widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\left(\boldsymbol{\phi}(\boldsymbol{\omega})\boldsymbol{\phi}(\boldsymbol{\omega})^\top + \sigma^2\mathbf{I}\right)\widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}$ denoted as μ_i for $i \geq 2$. Also, as explained before, the dominant term will be ($t = 2$), therefore we have that thus, using Eq. (S10) we have that

$$\begin{aligned}&\operatorname{tr}\left(\mathop{\mathbb{E}}_{\mathbb{P}(\boldsymbol{\omega})}\left[\log\left(\mathbf{I} - \left(\mathbf{I} - \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\widetilde{\mathbf{K}}_J\widehat{\mathbf{K}}_{\mathbf{XX}}^{-1/2}\right)\right)\right]\right) \\ &= \mathcal{O}(1/J)\end{aligned}$$

Therefore, we can conclude that

$$\mathop{\mathbb{E}}_{\mathbb{P}(\boldsymbol{\omega})}\left[\log\left|\widetilde{\mathbf{K}}_J\right|\right] - \log\left|\widehat{\mathbf{K}}_{\mathbf{XX}}\right| = \mathcal{O}(1/J).$$

□

B. Further Derivations of Randomized Truncation Estimators

Here we prove the unbiasedness of both Russian Roulette (RR) Eq. (6) and Single Sample (SS) Eq. (7) estimators. Recall from Sec. 2.3 that we wish to estimate the expensive

$$\psi = \textstyle\sum_{j=1}^{H}\Delta_j, \quad H \in \mathbb{N}\cup\{\infty\},$$

by computing just the first $J \ll H$ terms, where $J \in \{1,\ldots,H\}$ is randomly drawn from the truncation distribution $\mathbb{P}(\mathcal{J} = J)$. RR and SS estimators, denoted as $\bar{\psi}$, offer two strategies for up-weighting the J computed terms, which, as we will prove below, yield an unbiased estimator $\mathbb{E}_J[\bar{\psi}] = \psi$.

B.1. Unbiasedness of the RR Estimator

The RR estimator of Eq. (6) is unbiased, i.e., $\mathbb{E}_J\left[\bar{\psi}_J\right] = \psi$.

Proof.

$$\begin{aligned}\mathbb{E}_J\left[\bar{\psi}_J\right] &= \mathbb{E}_J\left[\sum_{j=1}^{J}\frac{\Delta_j}{\mathbb{P}(\mathcal{J}\geq j)}\right] \\ &= \mathbb{E}_J\left[\sum_{j=1}^{H}\frac{\Delta_j}{\mathbb{P}(\mathcal{J}\geq j)}\cdot\mathbb{I}_{j\leq J}\right] \\ &= \sum_{j=1}^{H}\frac{\Delta_j}{\mathbb{P}(\mathcal{J}\geq j)}\mathbb{E}_J[\mathbb{I}_{j\leq J}] \\ &= \sum_{j=1}^{H}\frac{\Delta_j}{\mathbb{P}(\mathcal{J}\geq j)}\left[\sum_{J=1}^{H}\mathbb{P}(\mathcal{J}=J)\cdot\mathbb{I}_{J\geq j}\right] \\ &= \sum_{j=1}^{H}\frac{\Delta_j}{\mathbb{P}(\mathcal{J}\geq j)}\mathbb{P}(\mathcal{J}\geq j) = \psi\end{aligned}$$

□

B.2. Unbiasedness of the SS Estimator

The SS estimator of Eq. (7) is unbiased, i.e., $\mathbb{E}_J\left[\bar{\psi}_J\right] = \psi$.

Proof.

$$\begin{aligned}\mathbb{E}_J\left[\bar{\psi}_J\right] &= \mathbb{E}_J\left[\sum_{j=1}^{H}\frac{\Delta_j}{\mathbb{P}(\mathcal{J}=j)}\cdot\mathbb{I}Jj1_{j=J}\right] \\ &= \sum_{j=1}^{H}\frac{\Delta_j}{\mathbb{P}(\mathcal{J}=j)}\mathbb{E}_J[\mathbb{I}_{j=J}] \\ &= \sum_{j=1}^{H}\frac{\Delta_j}{\mathbb{P}(\mathcal{J}=j)}\sum_{J=1}^{H}\mathbb{P}(\mathcal{J}=J)\cdot\mathbb{I}_{J=j} \\ &= \sum_{j=1}^{H}\frac{\Delta_j}{\mathbb{P}(\mathcal{J}=j)}\mathbb{P}(\mathcal{J}=j) = \psi\end{aligned}$$

□

B.3. Minimizing the Variance of the SS Estimator

Below we will derive the optimal distribution that minimizes the variance of our SS estimator. Note that for a given truncation distribution, we have that

$$\begin{aligned}\mathbb{V}_J\left(\bar{\psi}\right) &= \sum_{j=1}^{H}\frac{\Delta_j^2}{\mathbb{P}(\mathcal{J}=j)^2}\mathbb{V}_J\left(\mathbb{I}_{\mathcal{J}=j}\right) \\ &= \sum_{j=1}^{H}\Delta_j^2\left(\frac{1-\mathbb{P}(\mathcal{J}=j)}{\mathbb{P}(\mathcal{J}=j)}\right)\end{aligned}$$

since $\mathbb{I}_{\mathcal{J}=j}$ is a Bernoulli random variable with probability $\mathbb{P}(\mathcal{J}=j)$, we can plug-in its variance and derive the second equality. Hence, to find the truncation distribution that minimizes the variance of the SS estimator we can solve the following constraint optimization problem.

$$\min_{p} \sum_{j=1}^{H} \Delta_j^2 \frac{1-p_j}{p_j} \quad \text{s.t.} \quad \sum_{j=1}^{H} p_j = 1, \quad p_j \geq 0$$

where p_j is acting as a shorthand of $\mathbb{P}(\mathcal{J}=j)$. The Lagrangian of the problem is

$$\mathcal{L}(p,\lambda) = \sum_{j=1}^{H} \Delta_j^2 \frac{1-p_j}{p_j} + \lambda \left(1 - \sum_{j=1}^{H} p_j\right)$$

where we can ignore the nonnegativity constraints in p_j as long as $\Delta_j > 0$ (see solution below). Hence, the first order conditions dictate that

$$\frac{\partial \mathcal{L}}{\partial p_j}(p_j^\star, \lambda^\star) = -\left(\frac{\Delta_j^2}{p_j^\star}\right)^2 + \lambda^\star = 0$$

therefore, if we take the ratio of the probabilities with respect to the first we get that $p_j^\star = \frac{\Delta_j}{\Delta_1} p_1^\star$. If we substitute this expression in the equality constraint we get that

$$\mathbb{P}^\star(\mathcal{J}=j) = \frac{\Delta_j}{\sum_{i=1}^{H} \Delta_i} \propto \Delta_j \tag{S11}$$

We emphasize that this result is a guide for practical choices of the truncation distribution. It is impractical to compute it as it will require evaluating all the Δ_j for $j = 1, \dots, H$. However, if we posses an estimate or a theoretical bound on rate of decay of each Δ_j then our unnormalized truncation distribution should also decay at this rate to minimize variance.

B.4. SS estimator as Importance Sampling

Here we derive the SS estimator by importance sampling the quantity ψ with $\mathbb{P}(\mathcal{J}=J)$ as our proposal distribution.

$$\psi = \sum_{J=1}^{H} \Delta_J$$

Next, re-write the summation above as an expectation over the discrete uniform distribution $J \sim \mathcal{U}[1,H] = \frac{1}{H}$, $\forall J$:

$$= H \sum_{J=1}^{H} \frac{1}{H} \Delta_J,$$

We now introduce an alternative proposal distribution $J \sim \mathbb{P}(\mathcal{J}=J)$:

$$= H \sum_{J=1}^{H} \frac{\Delta_J}{H\mathbb{P}(\mathcal{J}=J)} \mathbb{P}(\mathcal{J}=J)$$

$$= \mathop{\mathbb{E}}_{J\sim\mathbb{P}(\mathcal{J}=J)} \left[\frac{\Delta_J}{\mathbb{P}(\mathcal{J}=J)}\right],$$

Approximating the final expectation using a single Monte Carlo sample results in the SS estimator.

C. Estimating the Marginal Log Likelihood from RR-CG

Recall from Sec. 4.1 that we use the Russian Roulette estimator in conjunction with conjugate gradients to compute an unbiased (stochastic) gradient of the log marginal likelihood. In this section, we briefly describe how to obtain an unbiased estimate of the log marginal likelihood itself.

The $\mathbf{y}^\top \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1}\mathbf{y}$ term can be estimated directly from the $\widehat{\mathbf{K}}_{\mathbf{XX}}^{-1}\mathbf{y}$ solve in Eq. (16). The $\log|\widehat{\mathbf{K}}_{\mathbf{XX}}|$ term is less straightforward, as the estimate from the CG byproducts (Eq. 8) isn't readily expressed as a summation. Instead, we apply the Russian Roulette estimator to the following telescoping series:

$$\log|\widehat{\mathbf{K}}_{\mathbf{XX}}| \approx \|\mathbf{z}\|^2 \mathbf{e}_1^T \left(\log \mathbf{T}_{\mathbf{z}}^{(N)}\right) \mathbf{e}_1 \tag{S12}$$

$$= \|\mathbf{z}\|^2 \mathbf{e}_1^T \left(\log \mathbf{T}_{\mathbf{z}}^{(1)}\right) \mathbf{e}_1 + \sum_{j=2}^{N} \Delta_j,$$

where $\Delta_j = \|\mathbf{z}\|^2 \mathbf{e}_1^T \left(\log \mathbf{T}_{\mathbf{z}}^{(j)} - \log \mathbf{T}_{\mathbf{z}}^{(j-1)}\right) \mathbf{e}_1$. We can therefore apply the Russian Roulette estimator with truncation $\mathbb{P}(J)$ to Eq. (S12). This telescoping series can be expensive to compute. Since it is unnecessary for gradient-based optimization, we introduce it only as a tool to analyze the RR-CG log marginal likelihood.

D. Optimal Truncation Distributions

D.1. Proof of Theorem 3

Proof. For Thm. 3 we have to combine the results of Thm. 1 with the optimal distribution for RR that is derived in Beatson & Adams (2019). We will only focus on v_J which that approximates $\log|\widehat{\mathbf{K}}_{\mathbf{XX}}|$ since the procedure is analogous for u_J which approximates $\mathbf{y}^\top \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1}\mathbf{y}$. Beatson & Adams (2019, Theorem 5.4) state that the optimal RR truncation distribution that maximizes the ROE is

$$\mathbb{P}^*(J \geq j) \propto \sqrt{\frac{\mathbb{E}\left[\|v_j\|^2\right]}{c(j) - c(j-1)}}$$

where $c(j)$ is the computational cost of evaluating v_j which is the j-th term being approximated through Russian

Roulette. In this case, the v_j are nonrandom and single-valued and the cost per iteration is constant with respect to j. Hence, we can conclude that

$$\mathbb{P}^*\left(J \geq j\right) \propto v_j = \mathcal{O}\left(C^{-2j}\right)$$

which implies that $\mathbb{P}^*\left(J \leq j\right) \propto 1 - \mathcal{O}\left(C^{-2j}\right)$ since the derivative of an exponential function is of the same form we have that $\mathbb{P}^*\left(J = j\right) \propto \mathcal{O}\left(C^{-2j}\right)$. □

D.2. Proof of Theorem 4

The strategy for Thm. 4 is to relate the kernel approximation using $J+1$ random Fourier features to the kernel approximation using J features for the two components of the log marginal likelihood: the data-fit term $\mathbf{y}^\top \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1}\mathbf{y}$ and the model complexity term $\log|\widehat{\mathbf{K}}_{\mathbf{XX}}|$. For $\mathbf{y}^\top \widehat{\mathbf{K}}_{\mathbf{XX}}^{-1}\mathbf{y}$ we create this connection through the Sherman-Morisson formula and for the $\log|\widehat{\mathbf{K}}_{\mathbf{XX}}|$ we use the matrix determinant lemma. Throughout these proofs, we will require some results regarding positive definite matrices that we add as remarks below. Before stating those remarks, we will first introduce some auxiliary notation.

$$\mathbf{K}_J := \frac{1}{J}\sum_{j=1}^{J}\boldsymbol{\phi}_j\boldsymbol{\phi}_j^\top$$

$$\left(J+1\right)\mathbf{K}_{J+1} = J\mathbf{K}_J + \boldsymbol{\phi}_{J+1}\boldsymbol{\phi}_{J+1}^\top$$

Note the difference between the $\widetilde{\mathbf{K}}_J$ term used throughout the paper which includes the $\sigma^2\mathbf{I}$ against the $\mathbf{K}_J$ term which only includes the RFF features. Moreover, to reduce clutter we define the following matrix

$$\mathbf{W} := \left(\frac{J}{J+1}\mathbf{K}_J + \sigma^2\mathbf{I}\right)$$

whose use will become evident throughout the proof.

Remark 1. *Given a positive definite matrix* $\mathbf{A}$ *we have that for any* $a \in (0,1)$ *and any* $b > 0$

$$\|a\mathbf{A} + b\mathbf{I}\|_2^2 \leq \|\mathbf{A} + b\mathbf{I}\|_2^2$$

If we express $\mathbf{A} = \mathbf{V}\mathbf{D}\mathbf{V}^\top$, where $\mathbf{V}$ is an orthogonal matrix and $\mathbf{D}$ is a diagonal matrix containing the positive eigenvalues of $\mathbf{A}$, then we have that $a\mathbf{A}+b\mathbf{I} = \mathbf{V}\left(a\mathbf{D}+b\mathbf{I}\right)\mathbf{V}^\top$. This implies that the eigenvalues of $\mathbf{A}+b\mathbf{I}$ are larger than those of $a\mathbf{A}+b\mathbf{I}$. An immediate consequence of this remark is that $\|\mathbf{W}\|_2^2 \leq \left\|\mathbf{K}_J + \sigma^2\mathbf{I}\right\|_2^2$ or that $\left\|\mathbf{W}^{-1}\right\|_2^2 \leq \left\|\Lambda^{-1} + \sigma^{-2}\right\|_2^2 \leq \sigma^{-4}$ where Λ contains the positive eigenvalues of $\mathbf{K}_J$.

Remark 2. *Given a positive definite matrix* $\mathbf{A}$ *we have that for any* $a \in (0,1)$*, any* $b > 0$ *and any vector* $\mathbf{x}$

$$\mathbf{x}^\top\left(a\mathbf{A}+b\mathbf{I}\right)^{-1}\mathbf{x} \leq \mathbf{x}^\top\left(\mathbf{A}+b\mathbf{I}\right)^{-1}\mathbf{x} \leq b^{-1}\mathbf{x}^\top\mathbf{x}$$

This remark follows by using the same diagonal decomposition as the one used in the previous remark. Hence, by noting that the eigenvalues of $\left(b\mathbf{I}\right)^{-1}$ are larger than those of $\left(a\mathbf{A}+b\mathbf{I}\right)^{-1}$ and this last matrix has larger eigenvalues than $\left(\mathbf{A}+b\mathbf{I}\right)^{-1}$ then the result follows.

Proof of Theorem 4. We start by showing the rate of decay for the Δ_j involving the data-fit terms. Note that we can express

$$\begin{aligned}
&\mathbf{y}^\top\left(\mathbf{K}_{J+1}+\sigma^2\mathbf{I}\right)^{-1}\mathbf{y} \\
&= \mathbf{y}^\top\left(\frac{J}{J+1}\mathbf{K}_J + \sigma^2\mathbf{I} + \frac{\boldsymbol{\phi}_{J+1}\boldsymbol{\phi}_{J+1}^\top}{J+1}\right)^{-1}\mathbf{y} \\
&= \mathbf{y}^\top\left(\mathbf{W} + \frac{\boldsymbol{\phi}_{J+1}\boldsymbol{\phi}_{J+1}^\top}{J+1}\right)^{-1}\mathbf{y}
\end{aligned} \tag{S13}$$

Applying Sherman-Morisson formula to the inverse of the R.H.S results in

$$\mathbf{W}^{-1} - \frac{\left(\boldsymbol{\phi}_{J+1}^\top\mathbf{W}^{-1}\right)^\top\left(\boldsymbol{\phi}_{J+1}^\top\mathbf{W}^{-1}\right)}{\left(J+1\right)+\boldsymbol{\phi}_{J+1}^\top\mathbf{W}^{-1}\boldsymbol{\phi}_{J+1}}$$

where the symmetry of $\mathbf{W}$ allows us to express the numerator as above. Substituting the previous result into Eq. (S13) and rearranging terms allows us to conclude that

$$\begin{aligned}
&\mathbf{y}^\top\left(\sigma^2\mathbf{I}+\mathbf{K}_J\right)^{-1}\mathbf{y} - \mathbf{y}^\top\left(\sigma^2\mathbf{I}+\mathbf{K}_{J+1}\right)^{-1}\mathbf{y} \\
&\leq \mathbf{y}^\top\left(\sigma^2\mathbf{I}+\frac{J}{J+1}\mathbf{K}_J\right)^{-1}\mathbf{y} - \mathbf{y}^\top\left(\sigma^2\mathbf{I}+\mathbf{K}_{J+1}\right)^{-1}\mathbf{y} \\
&= \frac{\left(\boldsymbol{\phi}_{J+1}^\top\mathbf{W}^{-1}\mathbf{y}\right)^2}{\left(J+1\right)+\boldsymbol{\phi}_{J+1}^\top\mathbf{W}^{-1}\boldsymbol{\phi}_{J+1}} \\
&\leq \frac{\left(\boldsymbol{\phi}_{J+1}^\top\mathbf{W}^{-1}\mathbf{y}\right)^2}{J+1} \\
&\leq \frac{\left\|\boldsymbol{\phi}_{J+1}\right\|_2^2\left\|\mathbf{W}^{-1}\mathbf{y}\right\|_2^2}{J+1} \\
&\leq \frac{\left\|\boldsymbol{\phi}_{J+1}\right\|_2^2\sigma^{-4}\|\mathbf{y}\|_2^2}{J+1}
\end{aligned} \tag{S14}$$

where the first inequality follows from Remark 2, the second inequality occurs since $\boldsymbol{\phi}_{J+1}^\top\mathbf{W}^{-1}\boldsymbol{\phi}_{J+1} > 0$, the third inequality results from applying Cauchy-Schwarz and the fourth stems from Remark 1. Finally, taking expectations in Eq. (S14) we can conclude that

$$\begin{aligned}
&\mathop{\mathbb{E}}_{\mathbb{P}(\boldsymbol{\omega})}\left[\mathbf{y}^\top\left(\sigma^2\mathbf{I}+\mathbf{K}_J\right)^{-1}\mathbf{y}\right] \\
&- \mathop{\mathbb{E}}_{\mathbb{P}(\boldsymbol{\omega})}\left[\mathbf{y}^\top\left(\sigma^2\mathbf{I}+\mathbf{K}_{J+1}\right)^{-1}\mathbf{y}\right] \\
&= \mathcal{O}\left(1/J\right).
\end{aligned}$$

We now move into the rate of decay of the Δ_j terms involving the model complexity terms. Note that we can express

$$\left|\mathbf{K}_{J+1} + \sigma^2\mathbf{I}\right| = \left|\frac{J}{J+1}\mathbf{K}_J + \sigma^2\mathbf{I} + \frac{\boldsymbol{\phi}_{J+1}\boldsymbol{\phi}_{J+1}^\top}{J+1}\right|$$

then by using the matrix determinant lemma we have that

$$\begin{aligned}\left|\mathbf{K}_{J+1} + \sigma^2\mathbf{I}\right| &= \left(1 + \frac{1}{J+1}\boldsymbol{\phi}_{J+1}^\top\mathbf{W}^{-1}\boldsymbol{\phi}_{J+1}\right)|\mathbf{W}| \\ &\leq \left(1 + \frac{1}{J+1}\boldsymbol{\phi}_{J+1}^\top\mathbf{W}^{-1}\boldsymbol{\phi}_{J+1}\right)\left|\mathbf{K}_J + \sigma^2\mathbf{I}\right| \\ &\leq \left(1 + \frac{\sigma^{-2}}{J+1}\boldsymbol{\phi}_{J+1}^\top\boldsymbol{\phi}_{J+1}\right)\left|\mathbf{K}_J + \sigma^2\mathbf{I}\right|\end{aligned} \tag{S15}$$

where the first inequality follows from Remark 1 and from noting that the determinant is equivalent to the product of the eigenvalues of the matrix. The second inequality follows from Remark 2. Now, we will take the logarithms and the expectations of Eq. (S15). We will first focus on the first term in the RHS. By using a Taylor expansion of the logarithm up to a second term we have that

$$\begin{aligned}&\mathop{\mathbb{E}}_{\mathbb{P}(\boldsymbol{\omega})}\left[\log\left(1 + \frac{\sigma^{-2}}{J+1}\boldsymbol{\phi}_{J+1}^\top\boldsymbol{\phi}_{J+1}\right)\right] \\ &= \frac{\sigma^{-2}\,\mathbb{E}_{\mathbb{P}(\boldsymbol{\omega})}\left[\boldsymbol{\phi}_{J+1}^\top\boldsymbol{\phi}_{J+1}\right]}{J+1} + \mathcal{O}\left(1/J^2\right)\end{aligned}$$

therefore, substituting the previous result into Eq. (S15)

$$\begin{aligned}&\mathop{\mathbb{E}}_{\mathbb{P}(\boldsymbol{\omega})}\left[\log\left|\mathbf{K}_{J+1} + \sigma^2\mathbf{I}\right|\right] - \mathop{\mathbb{E}}_{\mathbb{P}(\boldsymbol{\omega})}\left[\log\left|\mathbf{K}_J + \sigma^2\mathbf{I}\right|\right] \\ &= \frac{\sigma^{-2}\,\mathbb{E}_{\mathbb{P}(\boldsymbol{\omega})}\left[\boldsymbol{\phi}_{J+1}^\top\boldsymbol{\phi}_{J+1}\right]}{J+1} + \mathcal{O}\left(1/J^2\right) \\ &= \mathcal{O}\left(1/J\right)\end{aligned}$$

which concludes the analysis of the rate of decay of the log determinant terms. Thus, since each rate of decay if $\mathcal{O}\left(1/J\right)$, then following Eq. (S11) we have that the truncation distribution that minimizes the variance is

$$\mathbb{P}^*\left(J\right) \propto \Delta_J = \mathcal{O}\left(1/J\right).$$

□

E. Experiment Details.

We provide the experiment details for the predictive performance experiments in Sec. 5.

Experiment setup. To optimize the hyperparameters of the CG, RR-CG and Cholesky models, we use an Adam optimizer with learning rate = 0.01 and a MultiStepLR scheduler dropping the learning rate by a factor of 10 at the 50%, 70% and 90% of the optimization iterations. We run the optimization for 1500, 800 and 300 iterations on small (PoleTele, Elevators and Bike), medium (Kin40K, Protein, KEGG and KEGGU) and large (3DRoad) datasets, respectively. The number of iterations for CG and the expected number of iterations for RR-CG are both set to 100. The latter is achieved by using the truncation distribution from Eq. (17) with $\lambda = 0.05$ and $J_{\min} = 80$.

For CG and RR-CG, we use the rank-5 pivoted Cholesky preconditioner of Gardner et al. (2018). To reduce the number of optimization steps needed for the 3DRoad dataset, we initialize the hyperparameters to those found with the (biased) sgGP method. During evaluation, we compute the predictive mean using $1{,}000$ iterations of CG. Predictive variances are estimated using the rank-100 Lanczos approximation of Pleiss et al. (2018).

For RFF we use 1,000 random Fourier features across all experiments. In terms of optimization, we use an Adam optimizer with learning rate = 0.005 for KEGG, 0.001 for KEGGU and 0.01 for the remaining datasets. We also use a MultiStep scheduler that activates at 85%, 90%, 95% of the optimization iterations with a decay rate of 0.5 and also take a total of 500 optimization iterations on all the datasets.

In all the SVGP models we use $1{,}024$ inducing points, with a full-rank multivariate Gaussian variational distribution. Hyperparameters and variational parameters are jointly optimized for 300 epochs using minibatches of size $1{,}024$. As with the other baselines, we use the an Adam optimizer with a learning rate of 0.01, dropping the learning rate by a factor of 10 after 50%, 70% and 90% of the optimization iterations.

For sgGP, we train using minibatches of 16 data points. As suggested by Chen et al. (2020), the minibatches are constructed by sampling one training data point and selecting its 15 nearest neighbors. To accelerate optimization, we accumulate the gradients of $1{,}024$ minibatches before performing an optimization step (these $1{,}024$ minibatch updates can be performed in parallel, enabling GPU acceleration). We optimize the models for 300 epochs, using the same learning rate and scheduler as with SVGP. During evaluation, we use the same procedure as for CC/RR-CG ($1{,}000$ iterations of CG, rank-100 Lanczos variance estimates).

For POE, we divide the training dataset into disjoint subsets, each with $M = 1024$ data points. If N is not divisible by 1024, we pad the final subset with elements from the first subset. We train independent GP with independent hyperparameters on each of the subsets using the same optimization

Dataset	n	Cholesky	POE	RFF	RMSE SVGP	sgGP	CG	RR-CG
PolTele	9.6K	$.112 \pm .002$	$.174 \pm .006$	$\mathbf{.106 \pm .000}$	$.150 \pm .000$	$.128 \pm .001$	$.119 \pm .002$	$.112 \pm .002$
Elevators	10.6K	$\mathbf{.360 \pm .006}$	$.379 \pm .005$	$\mathbf{.365 \pm .001}$	$.376 \pm .006$	$\mathbf{.362 \pm .006}$	$\mathbf{.360 \pm .006}$	$\mathbf{.360 \pm .006}$
Bike	11.1K	$\mathbf{.035 \pm .003}$	$.071 \pm .002$	$.063 \pm .001$	$.045 \pm .002$	$1.044 \pm .334$	$\mathbf{.040 \pm .005}$	$\mathbf{.035 \pm .002}$
Kin40K	25.6K	—	$.248 \pm .001$	$\mathbf{.074 \pm .000}$	$.152 \pm .001$	$.081 \pm .000$	$.093 \pm .000$	$.091 \pm .001$
Protein	25.6K	—	$.715 \pm .007$	$\mathbf{.547 \pm .001}$	$.664 \pm .008$	$.562 \pm .005$	$\mathbf{.541 \pm .008}$	$\mathbf{.541 \pm .008}$
KEGG	31.2K	—	$.097 \pm .004$	$.101 \pm .001$	$\mathbf{.088 \pm .002}$	$\mathbf{.089 \pm .002}$	$.195 \pm .064$	$\mathbf{.087 \pm .003}$
KEGGU	40.7K	—	$.125 \pm .001$	$.129 \pm .001$	$.122 \pm .001$	$.123 \pm .001$	$\mathbf{.120 \pm .000}$	$\mathbf{.120 \pm .000}$
3DRoad	278K	—	$.675 \pm .001$	$.348 \pm .001$	$.439 \pm .002$	$.285 \pm .003$	$.202 \pm .003$	$\mathbf{.114 \pm .013}$

Dataset	n	Cholesky	POE	RFF	NLL SVGP	sgGP	CG	RR-CG
PolTele	9.6K	$-.464 \pm .006$	$-.252 \pm .010$	$-.159 \pm .005$	$-.442 \pm .004$	$\mathbf{-.480 \pm .004}$	$-.354 \pm .003$	$-.458 \pm .006$
Elevators	10.6K	$\mathbf{.425 \pm .013}$	$.469 \pm .016$	$.780 \pm .006$	$\mathbf{.442 \pm .015}$	$\mathbf{.426 \pm .015}$	$\mathbf{.429 \pm .012}$	$\mathbf{.425 \pm .013}$
Bike	11.1K	$-.984 \pm .018$	$-.799 \pm .010$	$.099 \pm .030$	$\mathbf{-1.514 \pm .024}$	85.715 ± 49.088	$-.971 \pm .008$	$-.982 \pm .018$
Kin40K	25.6K	—	$.464 \pm .002$	$1.407 \pm .004$	$\mathbf{-.410 \pm .003}$	$.427 \pm .001$	$.468 \pm .003$	$.449 \pm .013$
Protein	25.6K	—	$1.105 \pm .008$	$1.163 \pm .003$	$1.014 \pm .012$	$.951 \pm .004$	$\mathbf{.934 \pm .006}$	$\mathbf{.934 \pm .006}$
KEGG	31.2K	—	$-.874 \pm .011$	6.311 ± 2.323	$\mathbf{-1.022 \pm .023}$	$\mathbf{-.981 \pm .033}$	$.415 \pm .653$	$-.884 \pm .009$
KEGGU	40.7K	—	$-.636 \pm .002$	$4.000 \pm .303$	$\mathbf{-.685 \pm .005}$	$-.668 \pm .005$	$-.637 \pm .006$	$-.650 \pm .004$
3DRoad	278K	—	$1.031 \pm .002$	$1.317 \pm .004$	$\mathbf{.601 \pm .004}$	$.831 \pm .000$	$\mathbf{.613 \pm .010}$	$.776 \pm .030$

Table S1. Root-mean-square-error (RMSE) and negative log-likelihood (NLL) of exact GPs using CG, RRCG and other baselines on UCI regression datasets using a constant prior mean and a RBF kernel with independent lengthscale for each dimension. All trials were averaged over 3 trials with different splits. N and d are the size and dimensionality of the training dataset, respectively.

Dataset	n	Cholesky	POE	RFF	Training time (m) SVGP	sgGP	CG	RR-CG
PolTele	9.6K	$22.417 \pm .035$	$1.167 \pm .006$	$.464 \pm .002$	$3.862 \pm .018$	$.629 \pm .006$	$12.793 \pm .111$	$14.968 \pm .258$
Elevators	10.6K	$30.617 \pm .016$	$1.051 \pm .008$	$.503 \pm .002$	$4.236 \pm .023$	$.696 \pm .001$	$14.443 \pm .080$	$16.519 \pm .152$
Bike	11.1K	$34.991 \pm .016$	$1.364 \pm .012$	$.476 \pm .014$	$4.261 \pm .023$	$.691 \pm .002$	$11.634 \pm .131$	$13.669 \pm .080$
Kin40K	25.6K	—	$.930 \pm .006$	$.772 \pm .008$	$9.597 \pm .043$	$1.559 \pm .010$	$11.345 \pm .073$	$12.823 \pm .114$
Protein	25.6K	—	$.851 \pm .003$	$.716 \pm .008$	$11.115 \pm .033$	$1.867 \pm .007$	$10.507 \pm .044$	$12.142 \pm .052$
KEGG	31.2K	—	$1.135 \pm .003$	$.682 \pm .002$	$11.881 \pm .058$	$1.786 \pm .009$	$22.780 \pm .021$	$25.390 \pm .089$
KEGGU	40.7K	—	$1.140 \pm .005$	$.835 \pm .001$	$15.281 \pm .070$	$2.572 \pm .029$	$34.396 \pm .026$	$37.825 \pm .064$
3DRoad	278K	—	$2.176 \pm .022$	$6.089 \pm .031$	$104.164 \pm .294$	$22.615 \pm .147$	145.396 ± 1.373	$158.657 \pm .554$

Table S2. Total training time (in minutes) of exact GPs using CG, RRCG and other baselines on UCI regression datasets (see the number of optimization iterations for each method in experiment setup). All trials were averaged over 3 trials with different splits.

procedure as Cholesky models. Following Deisenroth & Ng (2015, Eqs. 11 and 12), the posterior distribution for a test input $\mathbf{x}^*$ is Gaussian with mean $\mu^*_{\text{POE}}(\mathbf{x}^*)$ and variance $\sigma^{2*}_{\text{POE}}(\mathbf{x}^*)$:

$$\mu^*_{\text{POE}}(\mathbf{x}^*) = \frac{\sigma^{2*}_{\text{POE}}(\mathbf{x}^*)}{K} \sum_{k=1}^{K} \sigma_k^{-2*}(\mathbf{x}^*) \mu_k^*(\mathbf{x}^*),$$

$$\sigma^{-2*}_{\text{POE}}(\mathbf{x}^*) = \frac{1}{K} \sum_{k=1}^{K} \sigma_k^{-2*}(\mathbf{x}^*),$$

where K is the number of independent GP experts and $\mu_k^*(\cdot)$ and $\sigma_k^{2*}(\cdot)$ are the posterior mean and variance of each expert model.

Full tables. In Table S1, we report the RMSE and NLL numbers which are used to plot Fig. 6 in the paper, and in Table S2 we report the corresponding training time.

F. Additional Experiments

F.1. Predictive Performance with Different CG iterations

Here we include an additional experiment to show that early truncated CG can be detrimental to GP learning while RR-CG remains robust to the expected truncation number.

We conduct GP learning with the CG, RR-CG and Cholesky methods in the Elevators dataset. We vary the number of (expected) iterations for CG and RR-CG from 20 to 100 and plot the corresponding RMSE / NLL in Fig. S1. From the figure, we conclude that: 1) early truncation of the CG algorithm impedes GP optimization and leads to poor predictions. This problem lessens as we increase the number of CG iterations from 20 to 100. 2) The RR-CG model is robust to the expected truncation number, as it keeps comparable RMSE and NLL values to the Cholesky model under different expected truncation numbers. This experiment can

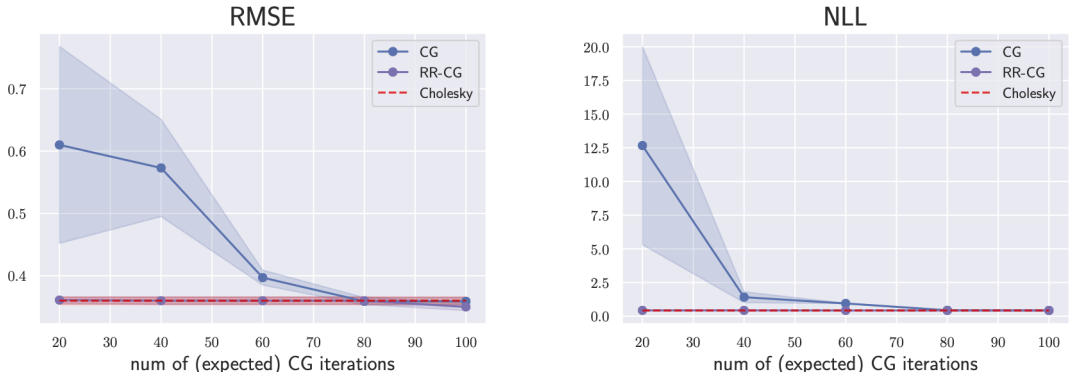

Figure S1. Predictive RMSE (left) and NLL (right) as a function of the number of (expected) CG iterations by optimizing Exact GP with CG (blue solid line) and RR-CG (purple solid line). Lower is better. The red dashed line corresponds to optimizing GP with Cholesky. The results are over three random seeds.

only be run in smaller datasets to be able to compare against Cholesky. We chose Elevators as this dataset requires the largest number of CG iterations before convergence hence making the difference of choosing between CG and RR-CG evident. We emphasize that RR-CG is a better alternative to early-truncated CG as the latter can perform poorly in some cases, whereas the former is robust to the choice of expected truncation number.

F.2. Convergence of GP hypeparameters for other datasets.

In this section, we show how SS-RFF and RR-CG converge to the Cholesky solution whereas the biased methods do not. We make this analysis for other datasets than the ones used in the main manuscript.

Using RFF with $1{,}000$ or $1{,}500$ features generates results that clearly diverge from the Cholesky solution and RFF with a $1{,}000$ features generates numerically unstable training. In contrast, SS-RFF with $1{,}500$ features is able to achieve the Cholesky solution after $1{,}000$ iterations. Nonetheless, SS-RFF with $1{,}000$ features also suffers from numerical instability as RFF $1{,}000$ but at a lesser degree.

The RR-CG models converge to optimal solutions, while the (biased) CG models diverge. Only the results for a low number of iterations is plotted since for more then 40 iterations CG and RR-CG are indistinguishable from Cholesky.

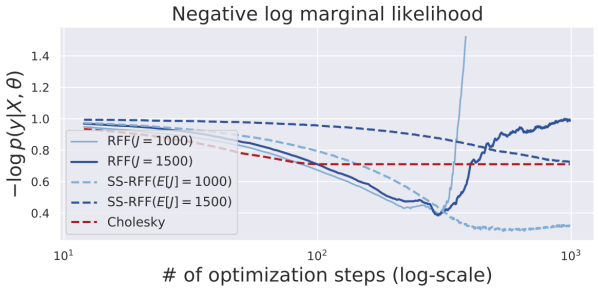

Figure S2. The GP optimization objective for models trained with RFF and SS-RFF. (Bike dataset, RBF kernel, Adam optimizer.) RFF models converge to sub-optimal log marginal likelihoods. SS-RFF models converge to (near) optimum values, yet require more than $100\times$ as many optimization steps.

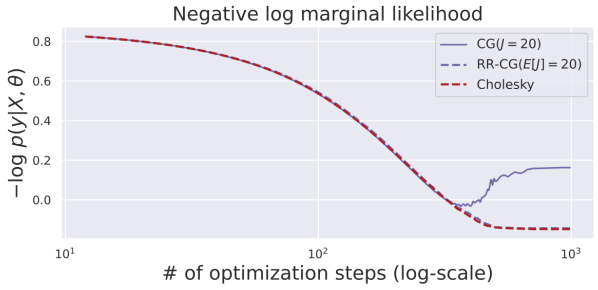

Figure S3. The GP optimization objective for models trained with CG and RR-CG. (PoleTele dataset, RBF kernel, Adam optimizer.) Models converge in < 100 steps of Adam.