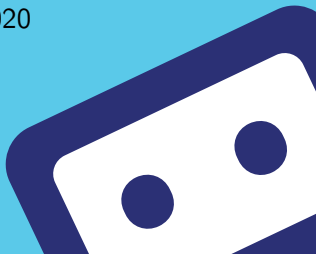
Sparse Gaussian Processes with Spherical Harmonic Features

Vincent Dutordoir[1], Nicolas Durrande[1] and James Hensman[2]

[1] PROWLER.io, [2]Amazon (Work completed while JH was at PROWLER.io)

International Conference of Machine Learning – 2020



Contribution

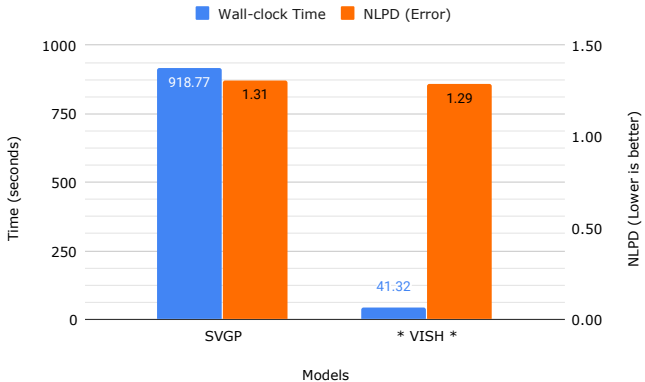
We improve the scaling of Sparse GPs with #datapoints and #inputs

Airline dataset:

- Regression problem
- $6{\cdot}10^6$ datapoints
- 8 input dimensions

Setup

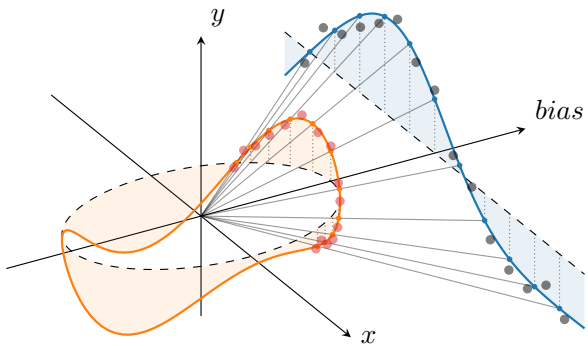
- GTX 1070 GPU



Variational Inference with Spherical Harmonics (VISH)

Gist of method:

- make inputs d + 1 dimensional
- project data radially on $\mathbb{S}^d$
- Fast SVGP on the sphere
- map predictions on $\mathbb{S}^d$ back to the original space

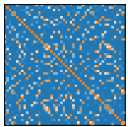


The efficiency of VISH comes from using *spherical harmonics as inducing functions* for the SVGP on the sphere.

From inducing points to inducing features

Inducing Points

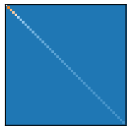
$u_m = f(z_m)$

$K_{uu} =$ 

K_{uu}^{-1} is $O(M^3)$

VISH

$u_m = \hbar f ; \ {}_m i_H$

$K_{uu} =$ 

K_{uu}^{-1} is $O(M)$

Orthogonality of the basisfunctions leads to diagonal K_{uu} and $O(M)$ inversion

Deep-dive

Sparse Variational Gaussian processes

Scalable and flexible

- Capture the GP by a set of inducing variables $\mathsf{u} = f(Z)$, at locations $\mathsf{z}_1; \cdots; \mathsf{z}_M$.

Sparse Variational Gaussian processes

Scalable and flexible

- Capture the GP by a set of inducing variables $\mathbf{u} = f(Z)$, at locations $z_1, \ldots, z_M$.
- Minimise KL-divergence from $p(f(\cdot)|y)$ to $q(f(\cdot)) = GP(\ (\cdot);\ (\cdot,\cdot'))$

$$
\begin{pmatrix}
(\cdot) = k_u^\top(\cdot) K_{uu}^{-1} \mathbf{m} \\
(\cdot,\cdot') = k(\cdot,\cdot') \quad k_u^\top(\cdot) K_{uu}^{-1} (K_{uu} \quad S) K_{uu}^{-1} k_u(\cdot')
\end{pmatrix},
$$

where $[K_{uu}]_{m,m'} = \mathrm{Cov}(u_m, u_{m'})$ and $[k_u(\cdot)]_m = \mathrm{Cov}(u_m, f(\cdot))$.

Sparse Variational Gaussian processes

Scalable and flexible

- Capture the GP by a set of inducing variables $\mathbf{u} = f(Z)$, at locations $z_1, \ldots, z_M$.
- Minimise KL-divergence from $p(f(\cdot)|y)$ to $q(f(\cdot)) = GP(\mu(\cdot), \Sigma(\cdot,\cdot))$

$$\begin{aligned} \mu(\cdot) &= k_u^\top(\cdot) K_{uu}^{-1} m \\ \Sigma(\cdot,\cdot') &= k(\cdot,\cdot') - k_u^\top(\cdot) K_{uu}^{-1} (K_{uu} - S) K_{uu}^{-1} k_u(\cdot') \end{aligned},$$

  where $[K_{uu}]_{m,m'} = \mathrm{Cov}(u_m, u_{m'})$ and $[k_u(\cdot)]_m = \mathrm{Cov}(u_m, f(\cdot))$.
- A more flexible (e.g. non-Gaussian likelihoods) and scalable (e.g. mini-batching) model at a cost of $O(M^3 + M^2 N)$.

Sparse Variational Gaussian processes

Scalable and flexible

- Capture the GP by a set of inducing variables $\mathrm{u} = f(Z)$, at locations $\mathrm{z}_1; : : : ; \mathrm{z}_M$.
- Minimise KL-divergence from $p(f(\ )jy)$ to $q(f(\ )) = GP(\ (\ ); \ (\ ; \ ^{0}))$

$$
\begin{pmatrix}
(\ ) = k_{\mathrm{u}}^{>}(\ )K_{\mathrm{uu}}{}^{1}\mathrm{m} \\
(\ ; \ ^{0}) = k(\ ; \ ^{0}) \quad k_{\mathrm{u}}^{>}(\ )K_{\mathrm{uu}}{}^{1}(K_{\mathrm{uu}} \quad S)K_{\mathrm{uu}}{}^{1}k_{\mathrm{u}}(\ ^{0})
\end{pmatrix};
$$

  where $[K_{\mathrm{uu}}]_{m;m^{0}} = \mathrm{Cov}(u_m; u_{m^{0}})$ and $[k_{\mathrm{u}}(\ )]_m = \mathrm{Cov}(u_m; f(\ ))$.
- A more flexible (e.g. non-Gaussian likelihoods) and scalable (e.g. mini-batching) model at a cost of $O(M^3 + M^2N)$.
- Speedup through structure in the K_{uu} matrix (e.g. Hensman et al 2017, VFF).

Outline

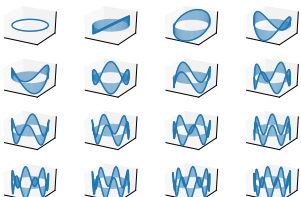
- Gaussian processes on the circle and hypersphere
- Spherical harmonics as inducing features
- Linear projection data on the hyper-sphere

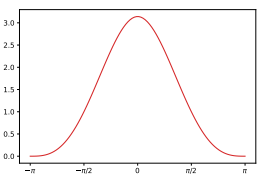
Gaussian processes on the circle

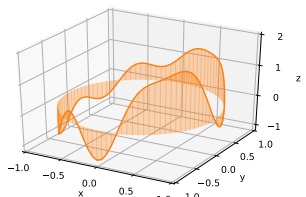
$$(\ ) = [\cos(i\ \ ); \sin(i\ \ )]_{i=0}^{1}$$

$$\mathrm{P}_{i=0}^{1}\ k(\ _1;\ _2) = \ \ _i\ \ _i(\ _1)\ \ _i(\ _2)$$

$$f = \mathrm{P}_{i}\ \ _i\ \ _i(\ ); \text{ with } \ \ _i \quad N(0;\ \ _i)$$

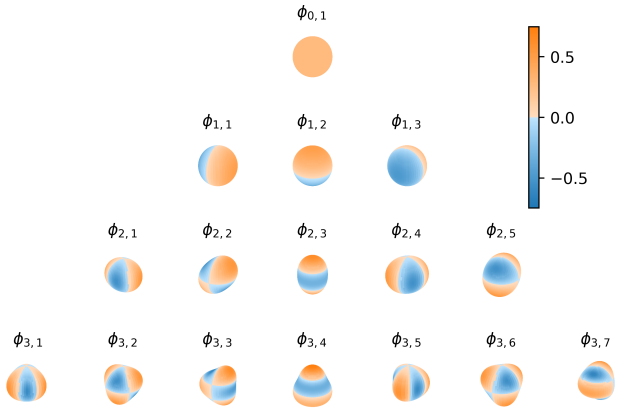






Spherical Harmonics

- Orthonormal basis on the hyper sphere
- Eigenfunctions the Laplace-Beltrami operator $\mathbb{S}^{d-1}\, _i = \, _i \, _i$
- Eigenfunction of zonal kernels



Mercer's theorem for zonal kernels on the sphere

- Zonal kernels are the spherical counterpart of stationary kernels $k(x; x^0) = k^0(\text{distance}(x; x^0))$.

Mercer's theorem for zonal kernels on the sphere

- Zonal kernels are the spherical counterpart of stationary kernels $k(x; x^0) = k^0(\text{distance}(x; x^0))$.
- Mercer's decomposition: Any zonal kernel k on the hypersphere can be decomposed as

$$k(x; x^0) = \sum_{i=0}^{\infty} \lambda_i \, e_i(x) \, e_i(x^0);$$

Mercer's theorem for zonal kernels on the sphere

- Zonal kernels are the spherical counterpart of stationary kernels $k(x; x') = k'(\text{distance}(x; x'))$.
- Mercer's decomposition: Any zonal kernel k on the hypersphere can be decomposed as

$$k(x; x') = \sum_{i=0}^{\infty} \quad _i \quad _i(x) \quad _i(x').$$

- Karhunen–Loève expansion: A GP f on the hypersphere with zonal covariance k can be written $f = \sum_{i=0}^{\infty} \quad _i \quad _i \quad _i$ with $_i \quad N(0; \quad _i)$:

Spherical harmonics as inducing features in SVGPs

- Define the kernel's RKHS H with reproducing inner-product:

$$\langle k(x; \cdot), h(\cdot) \rangle_H = h(x)$$

Spherical harmonics as inducing features in SVGPs

- Define the kernel's RKHS H with reproducing inner-product:

$$\langle k(x, \cdot), h(\cdot) \rangle_H = h(x)$$

- Approximate posterior constructed out of inducing features

$$\boxed{u_m = \langle f, \ _m \rangle_H}$$

Spherical harmonics as inducing features in SVGPs

- Define the kernel's RKHS H with reproducing inner-product:

$$\langle k(x;\ ); h(\ )\rangle_H = h(x)$$

- Approximate posterior constructed out of inducing features

$$\boxed{u_m = \langle f;\ _m\rangle_H}$$

$\Rightarrow$ Diagonal covariance matrix: $[K_{uu}]_{m;m'} = \mathrm{Cov}(u_m; u_{m'}) = \langle\ _m;\ _{m'}\rangle_H = \ ^1_m\ _{mm'}$

Spherical harmonics as inducing features in SVGPs

- Define the kernel's RKHS H with reproducing inner-product:

$$\langle k(x, \cdot), h(\cdot) \rangle_H = h(x)$$

- Approximate posterior constructed out of inducing features

$$\boxed{u_m = \langle f, \ _m \rangle_H}$$

$\Rightarrow$ Diagonal covariance matrix: $[K_{uu}]_{m;m'} = \mathrm{Cov}(u_m, u_{m'}) = \langle \ _m, \ _{m'} \rangle_H = \ _m^{-1} \ _{mm'}$

$\Rightarrow$ Spherical Harmonics as features $[k_u(\cdot)]_m = \mathrm{Cov}(u_m, f(\cdot)) = \ _m(\cdot)$

Spherical harmonics as inducing features in SVGPs

- Define the kernel's RKHS H with reproducing inner-product:

$$\langle k(x, \cdot), h(\cdot) \rangle_H = h(x)$$

- Approximate posterior constructed out of inducing features

$$\boxed{u_m = \langle f, \phi_m \rangle_H}$$

$\Rightarrow$ Diagonal covariance matrix: $[K_{uu}]_{m,m'} = \mathrm{Cov}(u_m, u_{m'}) = \langle \phi_m, \phi_{m'} \rangle_H = \lambda_m^{-1} \delta_{mm'}$

$\Rightarrow$ Spherical Harmonics as features $[k_u(\cdot)]_m = \mathrm{Cov}(u_m, f(\cdot)) = \phi_m(\cdot)$

$\Rightarrow$ A $O(M^2 N)$ approximate GP $q(f(\cdot))$

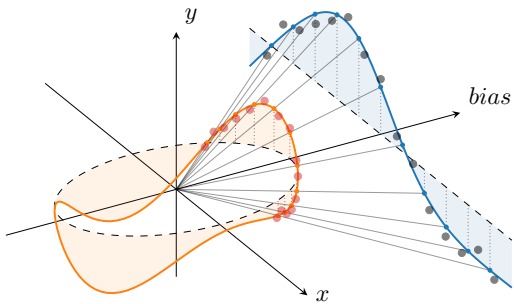
$$GP \quad ^{>}(\cdot)\, m; \quad k(\cdot, ^{0}) \quad ^{>}(\cdot)(\quad S) \quad (^{0}) \;;$$

where $= \mathrm{diag}(\ _1; \ldots; \ _M)$ and $(\cdot) = [\ _1(\cdot); \ldots; \ _M(\cdot)]$.

Linear mapping to the hypersphere

Most datasets do not correspond to data on a hypersphere...

The proposed solution is to augment the inputs with a constant variable (bias) before projecting it radially onto the hypersphere.

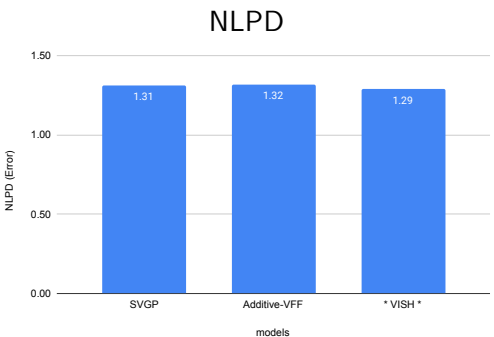


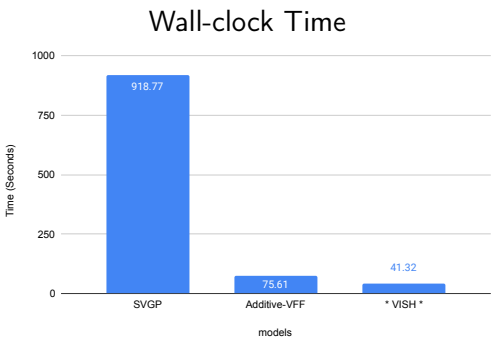
Although such construction may seem arbitrary, it is used implicitly in the Arc-Cosine kernel [Cho & Saul, 2009]:

$$k(\mathbf{x}, \mathbf{x}') = \underbrace{\|\mathbf{x}\|\|\mathbf{x}'\|}_{\text{radial}} \underbrace{(\sin\theta + (\pi - \theta)\cos\theta)}_{\text{angular}} \quad \text{with} \quad \theta = \arccos \frac{\mathbf{x}^\top \mathbf{x}'}{\|\mathbf{x}\|\|\mathbf{x}'\|}.$$

Experiment

Airline dataset: 6,000,000 datapoints regression task fitted in 40 seconds on a single cheap GTX 1070 GPU





Conclusion

Summary of the advantages

- It is the fastest SVGP model to date
  -) No need for expensive hardware
- The natural ordering of spherical harmonics makes our model scale nicely with the input dimension
  -) Does not suffer from the curse of dimensionality as VFF
- Similarities with Arc-cosine kernel makes extrapolation properties similar to Neural Networks

Reach out to have a chat if you want to know more!