Machine Learning stuff

Scalable GPs

GP suffers from cubic complexity so it’s gonna be long as shit to run

GP is a non-parametric statistical model, will be used for classification

Cubic because of the inversion and determinant of the nxn kernel matrix K\_nn = k(X,X)

This limits the scalability of GP and makes it unaffordable for large-scale datasets.

Hence, scalable GPs devote to improving the scalability of full GP while retaining favourable prediction quality for big data

2 main categories of scalable GPs: global and local approximations

Global: approximate the kernel matrix Knn through global distillation. Can be achieved by a true subset of the training data with m (m<<n) points resulting in the smaller matrix Kmm, the removal of uncorrelated entries in Knn, resulting in a sparse kernel matrix K(~ on top)nn with many 0 entries, and low-rank-representation measured between m inducing points and n training points (sparse approximations), resulting in the Nystro(..)m approximation Knn approx.=KnmK^(-1)mmKmn

Local: follows divide and conquer idea to focus on local subsets of training data. Only need to tackle a local expert with m0 (m0<<n) data points at each time. To produce smooth predictions with valid uncertainty, modelling averaging has been employed through mixture or product of experts.

For complexity comparisons, show figure 2 in paper.

Gaussian Process Classification(talk briefly about this as I’m not actually doing it)

Variational formulation can be exploited to allow classification in problems with millions of data points.

GPs provide priors over functions that can be used for machine learning tasks. Since this is classification, the posterior and marginal likelihood must be approximated.

Cost of inference is O(n^3) where n is number of data(yikes) sparse approaches reduce this, but rely on use of a series of inducing points which can be difficult to select. But there is a variational approach to optimize these points. This was extended to show how the variational objective could be reformulated with more parameters to enable stochastic optimization, allowing GPs to be fitted to millions of data (I only have 17911)

Sparse Multiclassifier

Generalising the binary classification Informative Vector Machine(IVM) to multiple classes

Achieves linear scaling for both number of classes and number of training points

IVM is a sparse approximation to Bayesian inference for binary GP classification models which combines Assumed Density Filtering(ADF)(or Bayesian on-line) projection updates with greedy forward selection of an active subset of the training sample using information-theoretic criteria from active learning. This can be extended to multiclass classification

This approach can be compared to multiclass extensions to SVM which is not based on a probabilistic model, rather employing a maximum margin discriminant. (this is ovo and ovr) which attempt a posthoc combination of a sequence of binary maximum margin discriminants trained on different binary splits of the training sample consistent with the targets. (ovo does every pair of classes, ovr does class vs all others (linear with increasing C)), I’m using ovr at the moment as I found one from GPy