Differentially Private Regression with Gaussian Processes

Michael T. Smith* **Mauricio A. Álvarez** **Max Zwiessele** **Neil D. Lawrence**
Department of Computer Science, University of Sheffield

Abstract

A major challenge for machine learning is increasing the availability of data while respecting the privacy of individuals. Here we combine the provable privacy guarantees of the differential privacy framework with the flexibility of Gaussian processes (GPs). We propose a method using GPs to provide differentially private (DP) regression. We then improve this method by crafting the DP noise covariance structure to efficiently protect the training data, while minimising the scale of the added noise. We find that this cloaking method achieves the greatest accuracy, while still providing privacy guarantees, and offers practical DP for regression over multi-dimensional inputs. Together these methods provide a starter toolkit for combining differential privacy and GPs.

1 Introduction

As machine learning algorithms are applied to an increasing range of personal data types, interest is increasing in mechanisms that allow individuals to retain their privacy while the wider population can benefit from inferences drawn through assimilation of data. Simple 'anonymisation' through removing names and addresses has been found to be insufficient [Sweeney, 1997, Ganta et al., 2008]. Instead, randomisation-based privacy methods (such as differential privacy, DP) provide provable protection against such attacks.

In this paper we investigate corrupting a Gaussian process's (GP's) fit to the data in order to make aspects of the training data private. Importantly this paper addresses the problem of making the training *outputs* ($\boldsymbol{y}$) of GP regression private, not its inputs. To motivate this, consider inference over census data. The inputs to our GP are the locations of every household in the country during a census. Note that the presence of a residential property need not be private. The values associated with that location (e.g. the person's religion or income) are private. Thus we can release the locations of the census households (in X) but protect the census answers (in $\boldsymbol{y}$). A second example is from a project in which the method has already been applied, analysing road collision data in Kampala, Uganda. Collision times and locations (which are public record) are entered in X, with the vehicles involved, the ages and genders of the victims, kept private (in $\boldsymbol{y}$). We are able to make differentially private inference around (for example) the times/places where children are most likely to be involved in collisions, providing useful insights, while ensuring that information about the victims is kept private.

We approach the problem of applying DP to GPs by finding a bound on the scale of changes to the GP's posterior mean function, in response to perturbations in the training outputs. We then use the results from Hall et al. [2013] to add appropriate Gaussian DP noise (Section 2). We find however that the added DP noise for this initial method is too large for many problems. To ameliorate this we consider the situation in which we know *a priori* the locations of the test points, and thus can reason about the specific correlation structure in the predictions for given perturbations in the training outputs (Section 3). Prior knowledge of the query is not unusual in methods for applying DP. Assuming the Gaussian mechanism is used to provide the DP noise, we are able to find the optimal noise covariance to protect training outputs. Finally we compare this strategy for inducing privacy with a DP query using the Laplace mechanism on bin means [Dwork and Roth, 2014, section 3.4], and show that it provides greater accuracy for a given privacy guarantee for our example dataset.
It is worth emphasising that we can still release the GP's covariance structure (as this only depends on the input locations, which we assume to be public) and the scale of the DP added noise. Thus the user is able to account for the uncertainty in the result. This paper combines the ubiquity of GP regression with the rigorous privacy guarantees offered by DP. This allows us to build a toolkit for applying DP to a wide array of problems amenable to GP regression.

Proceedings of the 21^{st} International Conference on Artificial Intelligence and Statistics (AISTATS) 2018, Lanzarote, Spain. PMLR: Volume 84.

*corresponding author: m.t.smith@sheffield.ac.uk

Related Work

A DP algorithm[Dwork et al., 2006, Dwork and Roth, 2014] allows privacy preserving queries to be performed by adding noise to the result, to mask the influence of individual data. This perturbation can be added at any of three stages in the learning process [Berlioz et al., 2015]: to the (i) training data, prior to its use in the algorithm, (ii) to components of the calculation (such as to the gradients or objective) or (iii) to the results of the algorithm. In this paper we focus on (iii) (adding the DP noise to the *predictions* in order to make aspects of the training data private). Considerable research has investigated the second of these options, in particular fitting parameters using an objective function which has been perturbed to render it differentially private [e.g. Chaudhuri et al., 2011, Zhang et al., 2012] with respect to the training data, or more recently, Song et al. [2013] described how one might perform stochastic gradient descent with DP updates. Some attention has also been paid to non-parametric models, such as histograms [Wasserman and Zhou, 2010] and other density estimators, such as the method described in Hall et al. [2013] which performs kernel density estimation (there are also parametric DP density estimators, such as those described by Wu et al. [2016] who use Gaussian mixtures to model density). For regression, besides using a perturbed objective function, one can also use the subsample-and-aggregate framework, as used by Dwork and Lei [2009, section 7], effectively protecting the parametric results of the regression. Heikkilä et al. [2017] use a similar idea for fitting parameters in a distributed, DP manner.

There are very few methods to perform DP non-parametric regression. To conduct a comparison we chose a binning method in which we make the data private by manipulating the bin means (using the Laplace mechanism) as other methods were not appropriate. For example, Chaudhuri et al. [2011], Rubinstein et al. [2012] and Song et al. [2013] were for classification, while Zhang et al. [2012] was for parametric models and only considered linear (and logistic) regression. It may be possible to extend their work, but this would be beyond the scope of the paper. Wasserman and Zhou [2010] use histogram queries. Machanavajjhala et al. [2008] use the less strict 'indistinguishability' definition. In summary, there is a dearth of methods for performing non-parametric differentially private regression. In particular there is an absence of research applying differential privacy to Gaussian processes (in Hall et al. [2013] they make use of GP's properties to provide DP to functions and vectors, but do not do the converse, making the GP's predictions private).

Differential Privacy

Briefly we reiterate the definition of differential privacy, from Dwork and Roth [2014]. To query a database in a differentially private manner, a randomised algorithm R is (ε, δ)-differentially private if, for all possible query outputs m and for all neighbouring databases D and D' (i.e. databases which only differ by one row),

$$P\Big(R(D) \in m\Big) \leq e^{\varepsilon} P\Big(R(D') \in m\Big) + \delta.$$

This says that we want each output value to be almost equally likely regardless of the value of one row: we do not want one query to give an attacker strong evidence for a particular row's value. ε puts a bound on how much privacy is lost by the query, with a smaller ε meaning more privacy. δ says this inequality only holds with probability $1 - \delta$.

2 Applying Differential Privacy to a Gaussian Process

The challenge is as follows; we have a dataset in which some variables (the inputs, $\boldsymbol{X}$) are public, for example the latitude and longitude of all homes in a country. We also have a variable we want to keep secret ($\boldsymbol{y}$, e.g. income). We want to allow people to make a prediction about this variable at a new location, while still ensuring that the dataset's secret variables remain private. In this section we fit a standard GP model to a dataset and calculate the bound on the scale of the perturbation we need to add to the posterior mean to provide a DP guarantee on the training outputs.

Hall et al. [2013] extended DP to functions. Consider a function, f, that we want to evaluate (with privacy guarantees). If the family of functions from which this function is sampled lies in a reproducing kernel Hilbert space (RKHS) then one can consider the function as a point in the RKHS. We consider another function, f', that has been generated using identical data except for the perturbation of one row. The distance, $||f - f'||$, between these points is bounded by the sensitivity, Δ. The norm is defined to be $||g|| = \sqrt{\langle g, g \rangle_H}$. Specifically the sensitivity is written $\Delta \geq \sup_{D \sim D'} ||f_D - f_{D'}||_H$. Hall et al. [2013] showed that one can ensure that a perturbation of f, $\widetilde{f}$, is (ε, δ)-DP by adding a (scaled) sample G from a Gaussian process prior (which uses the same kernel as f),

$$\widetilde{f} = f + \frac{\Delta \mathrm{c}(\delta)}{\varepsilon} G \tag{1}$$

where DP is achieved if

$$\mathrm{c}(\delta) \geq \sqrt{2 \log(2/\delta)} \tag{2}$$

Relating this to the definition of DP, finding Δ allows us to know how much the function f can change between neighbouring databases. We then choose the scale of the noise added by the randomised algorithm, M, to mask these changes.

We next extend these results, from Hall et al. [2013], to the predictions of a GP. In the GP case we have some data (at

inputs X and outputs $\boldsymbol{y}$). We assume for this paper that the *inputs* are non-private (e.g. people's ages), while the outputs are private (e.g. number of medications).

The mean function of a GP posterior lies in an RKHS. We need to add a correctly scaled sample to ensure its DP release. It will become clear that the covariance function does *not* need perturbation as it does not contain direct reference to the output values.

Using the notation of Rasmussen and Williams [2006], the predictive distribution from a GP at a test point $\boldsymbol{x}_*$ has mean $\bar{f}_* = \boldsymbol{k}_*^\top \left(K' + \sigma_n^2 I\right)^{-1} \boldsymbol{y}$, and variance $V[f_*] = k(\boldsymbol{x}_*, \boldsymbol{x}_*) - \boldsymbol{k}_*^\top \left(K' + \sigma_n^2 I\right)^{-1} \boldsymbol{k}_*$, where $\bar{f}_*$ is the mean of the posterior, $k(\boldsymbol{x}_*, \boldsymbol{x}_*)$ is the test point's prior variance, $\boldsymbol{k}_*$ is the covariance between the test and training points, K' is the Gram matrix describing the covariance between the training points, σ_n^2 is the variance of the iid noise added to each observation and $\boldsymbol{y}$ are the outputs observed values of the training data.

Ignoring any previous parameter selection, the variance does not depend on the training output values (in $\boldsymbol{y}$) so we only need to make the mean function private.

We can rewrite the above expression for the mean as the weighted sum of n kernel functions, $\bar{f}(\boldsymbol{x}_*) = \sum_{i=1}^n \alpha_i k(\boldsymbol{x}_i, \boldsymbol{x}_*)$, where $\boldsymbol{\alpha} = \left(K' + \sigma_n^2 I\right)^{-1} \boldsymbol{y}$. For simplicity in the following we replace $K' + \sigma_n^2 I$ with K, effectively combining a general kernel with a white-noise kernel. To apply the DP algorithm described by Hall et al. [2013] we need to find the (squared) distance in RKHS between the original and perturbed functions,

$$||f_D(\boldsymbol{x}_*) - f_{D'}(\boldsymbol{x}_*)||_H^2 = \left\langle f_D(\boldsymbol{x}_*) - f_{D'}(\boldsymbol{x}_*), f_D(\boldsymbol{x}_*) - f_{D'}(\boldsymbol{x}_*) \right\rangle_H. \quad (3)$$

In Hall et al. [2013, section 4.1], the vector $\boldsymbol{x}$ is identical to $\boldsymbol{x}'$ with the exception of the last element n. In our case the inputs are identical (we are not trying to protect this part of the data). Instead it is to the values of $\boldsymbol{y}$ (and hence $\boldsymbol{\alpha}$) that we need to offer privacy. To compute the norm in (3), we consider the effect a difference between $\boldsymbol{y}$ and (perturbed) $\boldsymbol{y}'$ has on the mean prediction function f_D at $\boldsymbol{x}_*$.

$$f_D(\boldsymbol{x}_*) - f_{D'}(\boldsymbol{x}_*) = \sum_{i=1}^n \alpha_i k(\boldsymbol{x}_*, \boldsymbol{x}_i) - \sum_{i=1}^n \alpha_i' k(\boldsymbol{x}_*, \boldsymbol{x}_i) = \sum_{i=1}^n (\alpha_i - \alpha_i')\, k(\boldsymbol{x}_*, \boldsymbol{x}_i), \quad (4)$$

where $\boldsymbol{\alpha} = K^{-1}\boldsymbol{y}$ and (the perturbed) $\boldsymbol{\alpha}' = K^{-1}\boldsymbol{y}'$. In the kernel density estimation example in Hall et al. [2013], all but the last term in the two summations cancel as the

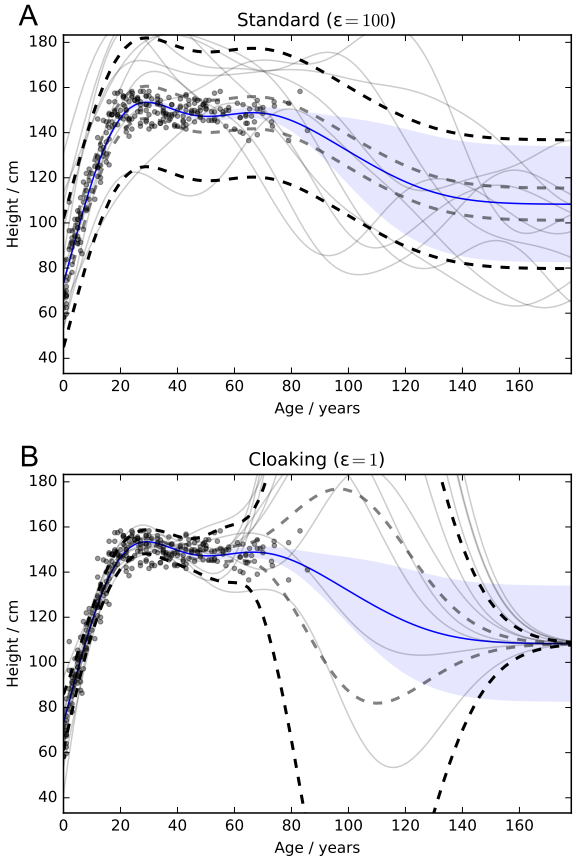

Figure 1: Heights and ages of female !Kung San. Figure A, standard GP method. Figure B, cloaking method. One can get an intuition for the utility of the DP output by considering the example DP samples. In Figure A one can see they deviate considerably from the actual mean, even with $\varepsilon = 100$. The cloaking method, using $\varepsilon = 1$, is able to provide reasonable predictions over the domain of the training data (although not in outlier regions). Solid blue lines, posterior means of the GPs; grey lines, DP samples; Black and grey dashed lines, SE and $\frac{1}{4}$SE confidence intervals for DP noise respectively; blue area, GP posterior variance (excluding noise). $\delta = 0.01$, $\Delta = 100$ cm.

α terms were absent. In our case however they remain and, generally, $\alpha_i \neq \alpha_i'$. We therefore need to provide a bound on the difference between the values of $\boldsymbol{\alpha}$ and $\boldsymbol{\alpha}'$. The difference between the two vectors is, $\boldsymbol{\alpha} - \boldsymbol{\alpha}' = K^{-1}(\boldsymbol{y} - \boldsymbol{y}')$. As K does not contain private information itself (it is dependent purely on the input and the features of the kernel) we can find a value for the bound using a specific K. See the supplementary material for a weaker, general, upper bound, for when the inputs are not known.

The largest change we protect against is a perturbation in *one* entry of $\boldsymbol{y}$ of at most $\pm d$. Therefore we assume that all the values of $\boldsymbol{y}$ and $\boldsymbol{y}'$ are equal except for the last element which differs by a value, which we will assume is a worst case of $\pm d$. Thus all the elements of $\boldsymbol{y} - \boldsymbol{y}'$ are

zero, except for the last, which equals $\pm d$. The result of multiplying K^{-1} with the vector $\boldsymbol{y} - \boldsymbol{y}'$ is the last column of K^{-1} scaled by $\pm d$. Equation 4 then effectively adds up the scaled last column, $\pm d[K^{-1}]_{:,n}$, but with each value scaled by the kernel's value at that point, $k(\boldsymbol{x}_*, \boldsymbol{x}_i)$. We initially assume that the kernel values are bound between -1 and 1 (not unreasonable, as many kernels, such as the exponentiated quadratic, have this property, if we normalise our data). Thus a worst case result of the sum is for the positive values in the scaled column, $\pm d[K^{-1}]_{:,n}$ to be multiplied by 1 and for the negative values to be multiplied by -1. Thus the largest result will be the sum of the last column's absolute (d-scaled) values. Finally, the *use of the last column is arbitrary*, so we can bound (4) by the maximum possible sum of any column's absolute values in K^{-1} (i.e. the infinity norm[1]), times d; i.e. $d||K^{-1}||_\infty$.

To reduce the scale of the DP noise a little more, we very briefly consider a slightly more restrictive case, that the value of the kernel is bound between $[0, 1]$. The bound is then calculated by finding the infinity norm for the following two matrices, and taking the larger. In one K^{-1} is modified so that all negative-values are ignored, in the other all values are initially negated, before the negative-values are discarded. We shall call this bound, $\mathrm{b}(K^{-1})$. The two options described are necessary to allow us to account for the uncertainty in the sign of $\boldsymbol{y} - \boldsymbol{y}'$, whose magnitude is bound by d but in an unknown direction. Returning to the calculation of the sensitivity, we can expand (4) and substitute into (3):

$$\begin{aligned} &||f_D(\boldsymbol{x}_*) - f_{D'}(\boldsymbol{x}_*)||_H^2 \\ &= \left\langle \sum_{i=1}^{n} (\alpha_i - \alpha_i') k(\boldsymbol{x}_*, \boldsymbol{x}_i), \sum_{i=1}^{n} (\alpha_i - \alpha_i') k(\boldsymbol{x}_*, \boldsymbol{x}_i) \right\rangle_H . \end{aligned} \tag{5}$$

To reiterate, we use our constraint that the chosen kernel has a range of 0 to 1, so the summations above will have a magnitude bounded by $d\,\mathrm{b}(K^{-1})$. This means that an upper bound on the sensitivity is, $||f_D(\boldsymbol{x}_*) - f'_D(\boldsymbol{x}_*)||_H^2 \leq d^2\ \mathrm{b}(K^{-1})^2$. If two training points (with differing output values) are very close together, the mean function (and thus the bound described above) can become arbitrarily large if the Gaussian noise term σ_n^2 is small. However in reality if very nearby points have different values then the underlying system presumably has some noise involved, which we model as additional Gaussian noise. Thus the off diagonals of K would remain smaller than the values on the diagonal, leading to a reasonable bound. In all the datasets examined so far, the selected Gaussian noise parameter has always been sufficiently large to avoid an excessively large bound. In general model selection for our DP GP will need to trade off between relying on single data points (i.e. low noise, causing the DP noise to be large) or relying on individual points less, due to the larger Gaussian noise term (making the non-DP prediction less accurate, but reducing the scale of the DP bound).

[1]The infinity norm of a symmetric square matrix is the maximum of the sums of the absolute values of the elements of rows (or columns); $\max_i \sum_j |M_{ij}|$

!Kung San women example

We use, as a simple demonstration, the heights and ages of 287 women from a census of the !Kung [Howell, N., 1967]. We are interested in protecting the privacy of their heights, but we are willing to release their ages. We have set the lengthscale *a priori*, to 25 years as from our prior experience of human development this is the timescale over which gradients vary.[2] We can find the empirical value of our sensitivity bound on the inverse covariance matrix, $\mathrm{b}(K^{-1})$ and the value of $c(\delta)$, from (2). Substituting in our given values in (1) we find that we should scale our GP samples by 28.53. Figure 1A shows that even with large ε the DP noise overwhelms the function we want to estimate (consider the spread of DP samples in the figure). It is worth noting that, if the sensitivity of the training data had been smaller (for example count or histogram data, with $\Delta = 1$) then this method could produce usable predictions at reasonable ε. In the following section we find we are able to considerably reduce the scale of the DP noise by insisting that we are given the test input points *a priori*.

3 The Cloaking Method

The method described in the previous section is limited to low-sensitivity datasets (i.e. those for which adding a single individual would not cause much change in the posterior mean, such as histogram/count data) due to the excessive scale of the noise added. We now introduce an alternative we refer to as *cloaking*, that allows a considerable reduction in the DP noise added but at the cost of needing to know the test point inputs *a priori*. We approach this new method by first reasoning about the direction (across test points) noise is added by the earlier (Section 2) method, and comparing its effect to the effect of modifying a training point. The sensitivity in the earlier methods needed to be quite high because the noise added (sampled from the *prior* covariance) is *not necessarily in the direction a perturbation in a training output would cause.*

Consider the simple case of two training and two test points, illustrated in figure 2. Subfigure B illustrates (with an ellipse) the shape of the noise added to the predictions *if we sample from the prior* (as in Section 2). Subfigures C-F illustrate changes caused by the perturbation of the

[2]Hyperparameters are all set *a priori*, but appear precise as the data outputs were normalised to have $\mu = 0$ and $\sigma = 1$. Kernel variance $\sigma^2 = 7.72^2$ cm^2, Gaussian white noise $\sigma_n^2 = 14^2$ cm^2, DP: $\delta = 0.01$, $\varepsilon = 50.0$, $\Delta = 100$ cm (enforced by rectifying all values to lie 50 cm of the mean).

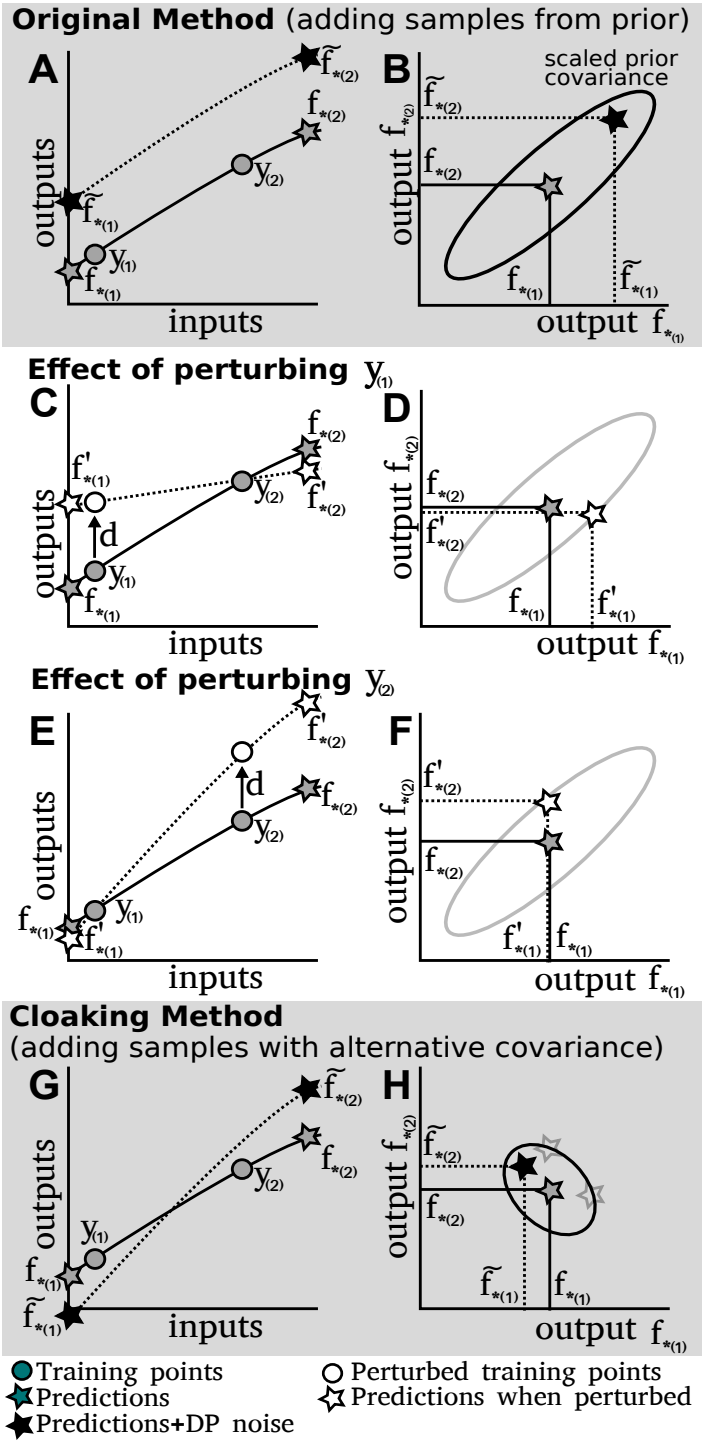

Figure 2: A illustrates the mechanism in Section 2 in which a scaled sample from the prior has been added to the test points $\boldsymbol{f}_*$ (grey stars) to produce DP private predictions (black stars) $\widetilde{\boldsymbol{f}}_*$ using (1). The model's long lengthscale means this moves the two test points up and down together (as they have a high covariance). Changing just one of the training points could not cause such changes in the test outputs. In B we plot the two test points against each other with the original predictions a grey star and the DP private predictions a black star. The covariance of the added DP noise is indicated with an ellipse.

In C we change just one of the training points, $y_{(1)}$, by adding a perturbation d to it. Using (8) we can see that test point $f_{*(1)}$ increased, while $f_{*(2)}$ decreased slightly. The two test points are plotted against each other in figure D. The grey ellipse indicates the covariance of the original method's noise. The new prediction is 'enclosed' by the covariance of the DP noise as the change must be indistinguishable from the DP noise.

In E and F we perturb the second training point, y_2 and plot the two test points against each other again. These figures demonstrate how changing single training points can not cause perturbations like those the original method adds, in figure A. The original method, by sampling from the scaled prior, is adding more noise than we need. Instead we should be sampling from a smaller covariance which only adds noise where necessary.

Figures G and H illustrating this alternative covariance (ellipse in H). A DP noise sample has been added, using (7), that is as small as possible by selecting the covariance using the cloaking mechanism, while still masking the possible perturbations one could cause by changing single training points. Note that the perturbed locations from figure D and F (indicated with faint grey stars) are enclosed by the new covariance ellipse.

training data to the predictions. The figure demonstrates that the prior does not provide the most efficient source of noise. In particular the large amount of correlated noise that is added in A and B is not necessary. Perturbations in individual training points cannot cause such correlated noise in the test outputs. To summarise; there is no perturbation in a single training point's output which could cause the predictions to move in the direction of the prior's covariance.

Differential Privacy for vectors of GP predictions

From Hall et al. [2013, proposition 3]: given a covariance matrix M and vectors of query results (in our case GP posterior mean predictions) $\mathbf{f}_*$ and $\mathbf{f}'_*$ from neighbouring databases D and D', we define the bound,

$$\sup_{D \sim D'} ||M^{-1/2}(\mathbf{f}_* - \mathbf{f}'_*)||_2 \leq \Delta \tag{6}$$

Δ is a bound on the scale of the prediction change, in term of its Mahalanobis distance with respect to the added noise covariance. The algorithm provides a (ε, δ)-DP output by adding scaled samples from a Gaussian distribution,

$$\tilde{\mathbf{f}}_* = \mathbf{f}_* + \frac{\mathrm{c}(\delta)\Delta}{\varepsilon} Z \tag{7}$$

where $Z \sim \mathcal{N}_d(0, M)$ and using function c from (2). We want M to have the greatest covariance in those directions most affected by changes in training points. We are able to compute K, the covariance between all training points (incorporating sample variance) and K_{*f} the covariance between training and test points. Given the training outputs $\mathbf{y}$, we can find the mean predictions for all test points simultaneously, $\mathbf{f}_* = K_{*f}K^{-1}\mathbf{y}$. The cloaking matrix $C = K_{*f}K^{-1}$ describes how the test points change wrt changes in training data. We use it to write the perturbed test values as

$$\mathbf{f}'_* = \mathbf{f}_* + C\left(\mathbf{y}' - \mathbf{y}\right). \tag{8}$$

We assume one training item i has been perturbed, by at most $\pm d$: $y'_i = y_i \pm d$. As y_i is the only training output value perturbed, we can see that the change in the predictions is dependent on only one column of C, $\mathbf{c}_i$; $\mathbf{f}'_* - \mathbf{f}_* = \pm d\mathbf{c}_i$ This can be substituted into the bound on Δ in (6). Rearranging the expression for the norm (and using M's symmetry);

$$\begin{aligned} ||dM^{-1/2}\mathbf{c}_i||_2 &= (dM^{-1/2}\mathbf{c}_i)^\top (dM^{-1/2}\mathbf{c}_i) \\ &= d^2\mathbf{c}_i^\top M^{-1}\mathbf{c}_i \end{aligned} \tag{9}$$

We want to find M such that the noise sample Z is small but also that Δ is minimised. A common way of describing that noise scale is to consider the determinant (generalised variance), the square root of the determinant (proportional to the volume of a confidence interval), or the log of the determinant (proportional to the differential entropy of a k-dimensional normal distribution (plus a constant) $\ln\left((2\pi e)^k\left|\mathbf{\Sigma}\right|\right)/2$. We use the latter but they will all have similar results. We show in the supplementary material that the optimal $M = \sum_i \lambda_i \mathbf{c}_i\mathbf{c}_i^\top$ with the $\boldsymbol{\lambda}$ found using gradient descent, $\partial L/\partial\lambda_j = \boldsymbol{c}_j^\top M^{-1}\boldsymbol{c}_j + 1$. We return to the example of the !Kung San women data to demonstrate the improvement in privacy efficiency. Figure 1B illustrates the results for a reasonable value of $\varepsilon = 1$.[3] The input domain is deliberately extended to demonstrate some features. First, where data is most concentrated the scale of the added noise is small. The effect one training point has there on the posterior prediction will be overwhelmed by its neighbours. A less intuitive finding is that the DP noise is greatest at about 110 years, far from the nearest data point. This is because the data's concentration acts like a pivot providing leverage to the outliers' effect on the posterior mean. Figure 2E partly demonstrates this, with the test point $f'_{*(2)}$ being changed slightly more than the perturbation in the training point. The third observation is that the added DP noise eventually approaches zero away from the training data; the posterior mean will equal the prior mean regardless of the training data's outputs. The RMSE without DP was 6.8cm and with $(1, 0.01)$-DP, 12.2cm, suggesting that this provides quite large, but practical levels of DP perturbation. The 200 test point inputs that make up the graph's predictions are exactly those specified as part of the cloaking algorithm (X_*); a large number of nearby test points does not degrade the quality of the predictions as they are all very highly correlated. The noise that is already added to ensure a test point does not breach DP, is almost exactly the noise that is needed by its close neighbours. As a further example we consider a spatial dataset of 4766 property sales since 2013 from the Land Registry [2017] in London.[4] Although this is a public

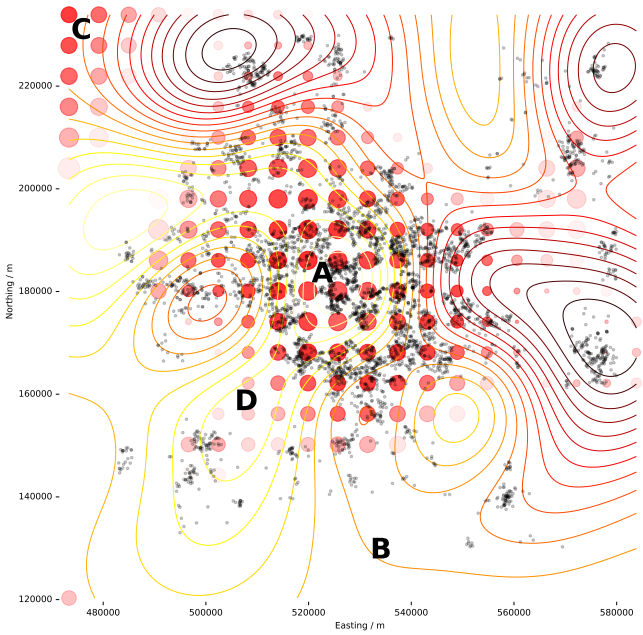

Figure 3: 4766 property prices in a $10,000$ km^2 square around London (2013-16). Dots, property locations; circles, DP price predictions at test points. Predicted price indicated by circle area. The scale of the DP noise indicated by transparency (Opaque: no DP noise. Transparent: DP noise std is at least 40% of Δ). Non-DP predictions indicated by contours from £215k to £437k. $\varepsilon = 1$ and $\delta = 0.01$. Areas (A) with high concentrations of training data have little DP noise, areas with few data have much more DP noise (B). Areas far from data return to GP's prior mean, and have little DP noise added (C). Interesting 'bridging' effects between data concentrations cause the DP noise to remain low as the posterior is 'supported' at both sides of areas with low density (e.g. D).

[3] EQ kernel, $\delta = 0.01$, lengthscale 25 years, Gaussian white noise 14 cm

[4] Thresholded to between £100k and £500k (so the sensitivity was bounded). $\varepsilon = 1$ and $\delta = 0.01$, lengthscale 15 km, Gaussian variance £^{2}400k^2.

dataset, one could imagine the outputs representing more private data (e.g. income or BMI). Figure 3 illustrates how the DP noise scale varies over the city. Training points are marked by dots. Test points are marked by the larger circles. Note how in areas where the training is concentrated the DP noise added is small, while in sparse areas the noise added is high. In the corners of the map, where there are no data, return to the GP prior's mean, and have little DP noise added.

Hyperparameter optimisation

So far in this paper we have selected the values of the kernel hyperparameters *a priori*. To demonstrate a DP method for their selection we evaluate the sum squared error (SSE) of k-fold cross-validation iterations for each hyperparameter configuration and select the optimum using the exponential mechanism (a method for selecting an item to maximise a utility, e.g. negative SSE, under DP constraints). Details of the method (including bounds on the sensitivity) can be found in the supplementary material.

4 Results

For comparison we binned the data (with added Laplace noise) to provide DP-predictions. Briefly, we bin the data, and add samples from the Laplace distribution with scale $\Delta f/\varepsilon$ where Δf is the bin sensitivity (equal to the maximum change possible in height, divided by the number of data points in that bin), then fit a GP to these, now noisy, bin values. We use both a simple Exponentiated Quadratic (EQ) kernel and a kernel that models a latent function which treats the observations as integrals over each bin. We found this often does better than simple binning when the data is noisy. We did not include the standard GP method (from Section 2) as we found it not competitive.

For the !Kung dataset, we applied the hyperparameter selection technique to the cloaking mechanism's outputs and compared it to the results of binning. We found that hyperparameter selection, for one or two parameters, caused a reduction in RMSE that was noticeable but not impractical. Specifically we used the exponential mechanism with the negative-SSE of 100-fold Monte-Carlo cross-validation runs to select hyperparameter combinations (testing lengthscales of between 3 and 81 years, and Gaussian noise variance of between 1.1^2 cm^2 and 12.7^2 cm^2), which we tested against a validation set to provide the expected RMSE. We fixed the DP ε to 1 for both the exponential mechanism and the cloaking and binning stages. The simple binning method (depending on bin size) had a RMSE of 23.4-50.7 cm, the integral method improved this to 14.2-20.8 cm. With no DP on parameter selection the cloaking method's RMSE would be 14.3 cm (comparable to the integral kernel's best bin size result). If we however select its hyperparameters using the exponential mechanism, the RMSE worsens to 17.4 cm. Thus there is a small, but manageable cost to the selection. Using data from the New York City bike sharing scheme, Citi Bike [2013][5] we predict journey time, given the latitude and longitude of the start and finish stations. The 4d Exponentiated Quadratic (EQ) kernel had lengthscales of between 0.02 and 0.781 degrees (latitude or longitude, equivalent to roughly 1.8 km to 75 km. $\sigma^2 = 1581^2$ s^2, $l = 0.05°$, $\sigma_n = 1605^2$ s^2. Durations were thresholded to a maximum of 2000 s.

We tested values of ε between 0.2 and 1.0 (with δ fixed at 0.01) and with DP disabled. Monte Carlo cross-validation was used to make predictions using the DP framework (4900 training, 100 test journeys). For comparison we binned the training data into between 81 and 10,000 bins, then computed DP means for these bins. These DP values were used as predictions for the test points that fell within that bin (those bins without training data were set to the population mean). Table 1 summarises the experimental results. The new cloaking function achieves the lowest RMSE, unless both ε and the lengthscales are small. With no DP the short-lengthscale cloaking method provides the most accurate predictions, as this is not affected by the binning and is capable of describing the most detail in the spatial structure of the dataset. The simple binning becomes less accurate with more bins, probably due to low occupancy rate (in a random training example, with 10,000 bins, only 11% were occupied, and of those 40% only had one point) and a form of overfitting. As $(1, 0.01)$-DP noise is added the simple-binning degrades quickly due to high DP noise added to low-occupancy bins. Xu et al. [2013] also describes a similar phenomenon. Predictions using the GP with an integral kernel fitted to these DP bin counts appears to provide some robustness to the addition of DP noise. As ε is further reduced, the cloaking method does better at longer lengthscales which allow more averaging over the training data. Simple binning becomes increasingly compromised by the DP noise.

5 Discussion

The cloaking method performs well, providing reasonably constrained levels of DP noise for realistic levels of privacy and provides intuitive features such as less DP-noise in those areas with the greatest concentration of training data. The earlier method, described in Section 2, required much more DP perturbation. For the cloaking method the lengthscale provides a powerful way of trading off preci-

[5] 163,000 subscribers, 600 stations located in a box bounded between latitudes $40.6794°$ and $40.7872°$, and longitudes $-74.0171°$ and $-73.9299°$. Unlike the house-price data we kept the locations in these global coordinates. Each fold of the Monte Carlo cross validation sampled 5000 rows from the 1,460,317 journeys in June 2016.

	lengthscale or bins	No DP	$\varepsilon = 1$	$\varepsilon = 0.5$	$\varepsilon = 0.2$
cloaking	$0.781°$	490 ± 14	493 ± 13	498 ± 13	525 ± 19
	$0.312°$	492 ± 15	497 ± 12	502 ± 17	545 ± 26
	$0.125°$	402 ± 7	437 ± 21	476 ± 17	758 ± 94
	$0.050°$	333 ± 11	434 ± 27	612 ± 78	1163 ± 147
	$0.020°$	314 ± 12	478 ± 22	854 ± 54	1868 ± 106
integral binning	10^4 bins	581 ± 5	586 ± 7	597 ± 12	627 ± 23
	6^4 bins	641 ± 6	640 ± 9	658 ± 17	736 ± 41
	3^4 bins	643 ± 6	649 ± 13	677 ± 22	770 ± 51
simple binning	10^4 bins	596 ± 12	1064 ± 69	1927 ± 191	4402 ± 434
	6^4 bins	587 ± 11	768 ± 58	1202 ± 206	2373 ± 358
	3^4 bins	550 ± 12	575 ± 24	629 ± 58	809 ± 110

Table 1: RMSE (in seconds, averaged over 30-fold X-validation, $\pm$ 95% CIs) for DP predictions of Citi Bike journey durations. Five lengthscales (in degrees latitude/longitude) and three bin resolutions were tested for the cloaking and binning experiments respectively. The cloaking method, with the right lengthscale, makes more accurate predictions than either of the binning methods. As we increase ε, cloaking needs longer lengthscales to remain competitive as this allows the predictions to 'average' over more training data.

sion in modelling spatial structure with the costs of increasing DP-noise. We could exploit this effect by using non-stationary lengthscales [e.g. Snoek et al., 2014, Herlands et al., 2015], incorporating fine lengthscales where data is common and expansive scales where data is rarefied. This could lead to DP noise which remains largely constant across the feature space. To further reduce the DP noise, we could manipulate the sample noise for individual output values. For the initial method, by adjusting the sample noise for individual elements we can control the infinity-norm. For the cloaking method, outlying training points, around the 'edge' of a cluster, could have their sample noise increased. One important issue is how to make DP GP predictions if we want to protect the values of the training *inputs*. This could be approached by considering a bound on the inverse-covariance function, a suggestion is provided in the supplementary material.

Future work is also needed to address how one optimises the hyperparameters of these models in a DP way. The method described in Section 3 works but is far from optimal. It may be possible to extend methods that use DP Bayesian optimisation to estimate values for hyperparameters [Kusner et al., 2015], or approximate the likelihood function to work with the method described in Han et al. [2014]. It would also be interesting to investigate how to modify the objective function to incorporate the cost of DP noise.

The actual method for releasing the corrupted mean function for the output-noise methods has not been discussed. Options include releasing a set of samples from the mean and covariance functions (a necessary step for the cloaking method, as the test-points need specifying in advance), or providing a server which responds with predictions given arbitrary input queries, sampling from the same Gaussian (for the cloaking method, querying new test points could be achieved by conditioning on the outputs given previously). The examples given here all use the EQ kernel but the cloaking method works with arbitrarily complex kernel structures and it has no requirement that the covariance function be stationary. GP classification is also an obvious next step.

Finally, in unpublished work, we have found evidence that the use of inducing inputs can significantly reduce the sensitivity, and thus DP noise required by reducing the predictions' dependency on outliers. Future work should be undertaken to investigate the potential for further reducing the DP noise through the use of inducing inputs.

We have presented novel methods for combining DP and GPs. In the longer term we believe a comprehensive set of methodologies could be developed to enhance their applicability in privacy preserving learning. We have applied DP for functions to GPs and given a set of known test points we were able to massively reduce the scale of perturbation for these points by considering the structure of the perturbation sensitivity across these points. In particular we found that the cloaking method performed considerably more accurately than the binning alternatives.

Acknowledgements

This work has been supported by the Engineering and Physical Research Council (EPSRC) Research Project EP/N014162/1.

References

Arnaud Berlioz, Arik Friedman, Mohamed Ali Kaafar, Roksana Boreli, and Shlomo Berkovsky. Applying differential privacy to matrix factorization. In *Proceedings of the 9th ACM Conference on Recommender Systems*, pages 107–114. ACM, 2015.

Kamalika Chaudhuri and Staal A Vinterbo. A stability-

based validation procedure for differentially private machine learning. In *Advances in Neural Information Processing Systems*, pages 2652–2660, 2013.

Kamalika Chaudhuri, Claire Monteleoni, and Anand D Sarwate. Differentially private empirical risk minimization. *Journal of Machine Learning Research*, 12(Mar): 1069–1109, 2011.

Citi Bike. Citi bike system data. `https://www.citibikenyc.com/system-data`, 2013.

Cynthia Dwork and Jing Lei. Differential privacy and robust statistics. In *Proceedings of the forty-first annual ACM symposium on Theory of computing*, pages 371–380. ACM, 2009.

Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3-4):211–407, 2014.

Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography Conference*, pages 265–284. Springer, 2006.

Srivatsava Ranjit Ganta, Shiva Prasad Kasiviswanathan, and Adam Smith. Composition attacks and auxiliary information in data privacy. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 265–273. ACM, 2008.

Rob Hall, Alessandro Rinaldo, and Larry Wasserman. Differential privacy for functions and functional data. *The Journal of Machine Learning Research*, 14(1):703–727, 2013.

Shuo Han, Ufuk Topcu, and George J Pappas. Differentially private convex optimization with piecewise affine objectives. In *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*, pages 2160–2166. IEEE, 2014.

Mikko Heikkilä, Yusuke Okimoto, Samuel Kaski, Kana Shimizu, and Antti Honkela. Differentially private Bayesian learning on distributed data. In *Advances in neural information processing systems*, 2017.

William Herlands, Andrew Wilson, Hannes Nickisch, Seth Flaxman, Daniel Neill, Wilbert Van Panhuis, and Eric Xing. Scalable Gaussian processes for characterizing multidimensional change surfaces. *arXiv preprint arXiv:1511.04408*, 2015.

Howell, N. Data from a partial census of the !Kung San, Dobe. 1967-1969. `https://public.tableau.com/profile/john.marriott#!/vizhome/kung-san/Attributes`, 1967.

Matt Kusner, Jacob Gardner, Roman Garnett, and Kilian Weinberger. Differentially private Bayesian optimization. In *International Conference on Machine Learning*, pages 918–927, 2015.

Land Registry. HM land registry price paid data. `https://data.gov.uk/dataset/land-registry-monthly-price-paid-data`, 2017.

Ashwin Machanavajjhala, Daniel Kifer, John Abowd, Johannes Gehrke, and Lars Vilhuber. Privacy: Theory meets practice on the map. In *Data Engineering, 2008. ICDE 2008. IEEE 24th International Conference on*, pages 277–286. IEEE, 2008.

Carl Edward Rasmussen and Christopher KI Williams. *Gaussian processes for machine learning*, volume 1. MIT press Cambridge, 2006.

Benjamin IP Rubinstein, Peter L Bartlett, Ling Huang, and Nina Taft. Learning in a large function space: Privacy-preserving mechanisms for SVM learning. *Journal of Privacy and Confidentiality*, 4(1):65–100, 2012.

Jasper Snoek, Kevin Swersky, Richard S Zemel, and Ryan P Adams. Input warping for Bayesian optimization of non-stationary functions. In *ICML*, pages 1674–1682, 2014.

Shuang Song, Kamalika Chaudhuri, and Anand D Sarwate. Stochastic gradient descent with differentially private updates. In *Global Conference on Signal and Information Processing (GlobalSIP), 2013 IEEE*, pages 245–248. IEEE, 2013.

Latanya Sweeney. Weaving technology and policy together to maintain confidentiality. *The Journal of Law, Medicine & Ethics*, 25(2-3):98–110, 1997.

James M Varah. A lower bound for the smallest singular value of a matrix. *Linear Algebra and its Applications*, 11 (1):3–5, 1975.

Larry Wasserman and Shuheng Zhou. A statistical framework for differential privacy. *Journal of the American Statistical Association*, 105(489):375–389, 2010.

Yuncheng Wu, Yao Wu, Hui Peng, Juru Zeng, Hong Chen, and Cuiping Li. Differentially private density estimation via Gaussian mixtures model. In *Quality of Service (IWQoS), 2016 IEEE/ACM 24th International Symposium on*. IEEE, 2016.

Jia Xu, Zhenjie Zhang, Xiaokui Xiao, Yin Yang, Ge Yu, and Marianne Winslett. Differentially private histogram publication. *The VLDB Journal*, 22(6):797–822, 2013.

Jun Zhang, Zhenjie Zhang, Xiaokui Xiao, Yin Yang, and Marianne Winslett. Functional mechanism: regression analysis under differential privacy. *Proceedings of the VLDB Endowment*, 5(11):1364–1375, 2012.