Foundations of Bayesian Learning from Synthetic Data

A.1 The β-Divergence

The β-divergence (βD) was fist introduced by Basu et al. (1998) under the name 'density power divergence', as a robust and efficient alternative to frequentist maximum-likelihood estimation. It has since been used to produce Bayesian posteriors, firstly in Ghosh and Basu (2016) before being unified by generalised Bayesian updating (Bissiri et al., 2016) in Jewson et al. (2018). Recent applications of the βD are vast, representing a growing area of research.

The β-divergence between two distributions with densities g and f with dominating measure μ (which we in general assume to be the Lebesque measure) is

$$\beta\mathrm{D}(g \,\|\, f) := \frac{1}{\beta+1}\int f^{\beta+1}d\mu - \frac{1}{\beta}\int f^{\beta}g d\mu + \frac{1}{\beta(\beta+1)}\int g^{\beta+1}d\mu. \tag{A.1}$$

When minimising $\beta\mathrm{D}(g \,\|\, f_\theta)$ for θ, the final term $\frac{1}{\beta(\beta+1)}\int g^{\beta+1}d\mu$ can be ignored and therefore

$$\theta_G^{\beta\mathrm{D}} := \mathrm{argmin}_{\theta\in\Theta}\beta\mathrm{D}(g \,\|\, f) = \mathrm{argmin}_{\theta\in\Theta}\mathbb{E}_{z\sim g}\left[\ell^{(\beta)}(z, f_\theta)\right], \tag{A.2}$$

with the law of large numbers providing that only a sample $\{x_i\}_{1:n} \sim g$ is needed to instantiate such a minimiser

$$\mathbb{E}_{z\sim g}\left[\ell^{(\beta)}(z, f_\theta)\right] \xleftarrow[n\to\infty]{} \frac{1}{n}\sum_{i=1}^{n}\ell^{(\beta)}(z_i, f_\theta), \quad x_i \sim g, \tag{A.3}$$

and from Eq. (9) of the main paper

$$\ell^{(\beta)}(z, f_\theta) := \frac{1}{\beta+1}\int f_\theta(y)^{\beta+1}dy - \frac{1}{\beta}f_\theta(z)^{\beta}.$$

The fact that

$$\lim_{\beta\to 0}\frac{1}{\beta}x^{\beta} = \log x, \tag{A.4}$$

is then enough to prove that $\lim_{\beta\to 0}\beta\mathrm{D} = \mathrm{KLD}$.

A.1.1 The robustness of the βD

This section provides some illustrations demonstrating the robustness of the general Bayesian update using $\ell_w(x, f_\theta)$ and $\ell^{(\beta)}(x, f_\theta)$ compared with traditional Bayesian updating (using $\ell_0(x, f_\theta)$).

Firstly we consider how observations are 'downweighted' compared with the prior under different learning / updating procedures. First, we let the model be a Gaussian location scale model $f_\theta = \mathcal{N}(\mu, \sigma^2)$. We start with a conjugate Normal-Inverse-Gamma (NIG) prior, $NIG(\mu, \sigma^2; a_0, b_0, \mu_0, \kappa_0) = \mathcal{IG}(\sigma^2; a_0, b_0) \times \mathcal{N}(\mu; \mu_0, \sigma^2/\kappa_0)$ with $(a_0, b_0, \mu_0, \kappa_0) = (2, 1, 0, 1/2)$ and consider the posterior after observing an observation 'in agreement' with the prior $x_{in} = 0.5$ and one not in agreement with the prior $x_{out} = 5$. Figure A.1, plots the prior and posterior predictives' densities after observing one observation in these two cases.

After seeing an 'inlying' observation consistent with the prior, all three methods learn similarly, with their posterior predictives' modes being shifted towards the observation. However, we see that using either $\ell_w(x, f_\theta)$ or $\ell^{(\beta)}(x, f_\theta)$ gives more relative weight to the prior than traditional Bayesian updating, as they continue to produce larger posterior variances driven by the prior. After seeing an 'outlier' the three methods produce very different inferences. One outlying observation can be seen to move the traditional Bayesian inference away from the prior predictive; the same effect is witnessed when using $\ell_w(x, f_\theta)$, although to a lesser extent in that the posterior predictive also carries a large variance compared to the traditional Bayesian one. Inference under the βD is very different given an outlier. The posterior predictive mode stays in agreement with the prior but the right tail of the posterior is heavier than the left in order to 'acknowledge' the outlying observation.

Next we formalise the influence (Kurtek and Bharath, 2015; Jewson et al., 2018) given to different observations under the different learning methods, $\ell_w(x, f_\theta)$ and $\ell^{(\beta)}(x, f_\theta)$ w.r.t the Gaussian model $f_\theta = \mathcal{N}(\mu, \sigma^2)$. Here we examine $R(\pi^{(\ell}(\theta|z_{1:n}, x), \pi^{\ell}(\theta|z_{1:n}))$: the Fisher-Rao metric between the general Bayesian posterior based on observations $\{z_{1:n}, x\}$ and the general Bayesian posterior based only on $z_{1:n}$, providing an idea of how an observation at x influences the posterior. Figure A.1 shows this for variable x, loss functions $\ell_w(x, f_\theta)$ and $\ell^{(\beta)}(x, f_\theta)$ for varying w and β, and $z_{1:n} \sim \mathcal{N}(0, 1)$ with $n = 200$. The influence plots under $\ell_w(x, f_\theta)$ are monotonically increasing, showing that as an observation becomes less likely under the current inference its influence over the analysis increases. Decreasing $w < 1$ decreases the influence of a new observation, but we can see this happens uniformly meaning an outlier is downweighted by the same amount as an observation near the current posterior mode. Under $\ell^{(\beta)}(x, f_\theta)$ the influence curves are no longer monotonic as the observation moves away from the current posterior mean. Initially, the influence of observations increases, mimicking inference under $\ell_w(x, f_\theta)$, but then after a point the influence starts to decrease as these observations become increasingly unlikely given the current inference. This allows βD-inference to adaptively reject the influence of outliers.

Lastly we show how the downweighting of the influence of observations illustrated above affects inference for large samples. We consider inference for a Gaussian model $f_\theta = \mathcal{N}(\mu, \sigma^2)$ based on two datasets of size $n = 1000$ generated from $g_1(x) = 0.9\mathcal{N}(0, 1^2) + 0.1\mathcal{N}(5, 3^2)$ and $g_2(x) = \mathcal{L}(0, 1)$. Generating process g_1 is referred to as an ϵ-contamination, where the model is correct for $(1 - \epsilon)\%$ of observations but is contaminated with $\epsilon\%$ outliers, whilst g_2 has heavier tails that f_θ. Figure A.2 plots the posterior predictive approximation of both g_1 and g_2 for traditional Bayesian updating and general Bayesian updating with $\ell_w(x, f_\theta)$ and $\ell^{(\beta)}(x, f_\theta)$. Firstly, for $n = 1000$ there is little difference between the traditional Bayesian inference and the general Bayesian inference using $\ell_w(x, f_\theta)$. Additionally we see that minimising the βD allows the general Bayesian inference to be less concerned with correctly capturing the tails of g_1 and g_2 and as a result allows it to provide a more accurate approximation to their modes.

A.1.2 The βD and optimising the learning trajectory

In Section 3.2 of the paper we state that we believe the βD minimising approximation to $\mathcal{G}_{\varepsilon,\delta}$ will often be a better approximation of F_0 than the KLD minimising approximation, as a result of its general robustness properties outlined above. That is to say that

$$D\left(F_0 \,\|\, f_{\theta^{\beta\mathrm{D}}_{\mathcal{G}_{\varepsilon,\delta}}}\right) < D\left(F_0 \,\|\, f_{\theta^{\mathrm{KLD}}_{\mathcal{G}_{\varepsilon,\delta}}}\right). \tag{A.5}$$

Sadly, proving any general results to this effect is difficult as a result of the intractability of the both of the βD minimising parameter and the data generating density of many popular synthetic data generators. Instead, to further justify this claim here, we provide brief experiments focusing on the Laplace mechanism S-DGP. In

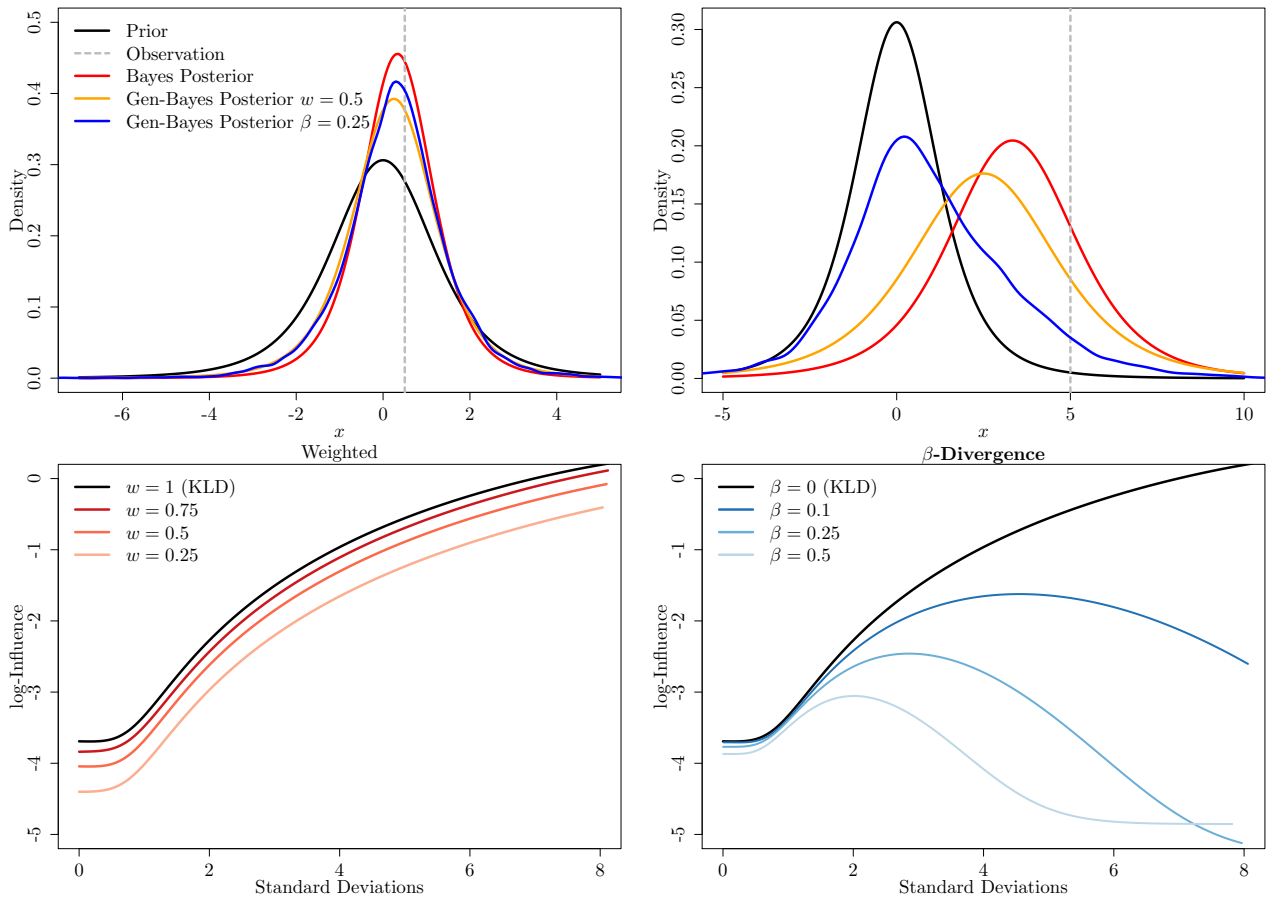

Figure A.1: Influence of Outliers. **Top**: NIG Prior predictive (**black**) and posterior predictives using a Gaussian model, $f_\theta = \mathcal{N}(\mu, \sigma^2)$, under traditional Bayesian updating ($\ell_0(x, f_\theta)$) (**red**) and general Bayesian updating with $\ell_w(x, f_\theta)$ (**orange**) and $\ell^{(\beta)}(x, f_\theta)$ (**blue**) after an inlying (**Left**) and outlying (**Right**) observation (**grey**). **Bottom**: log-Fisher-Rao-metric (Kurtek and Bharath, 2015) between the general Bayesian posterior with or without one observation at different posterior standard deviations away from the previous posterior mean for $\ell_w(x, f_\theta)$ (**Left**) and $\ell^{(\beta)}(x, f_\theta)$ (**Right**) under model $f_\theta = \mathcal{N}(\mu, \sigma^2)$.

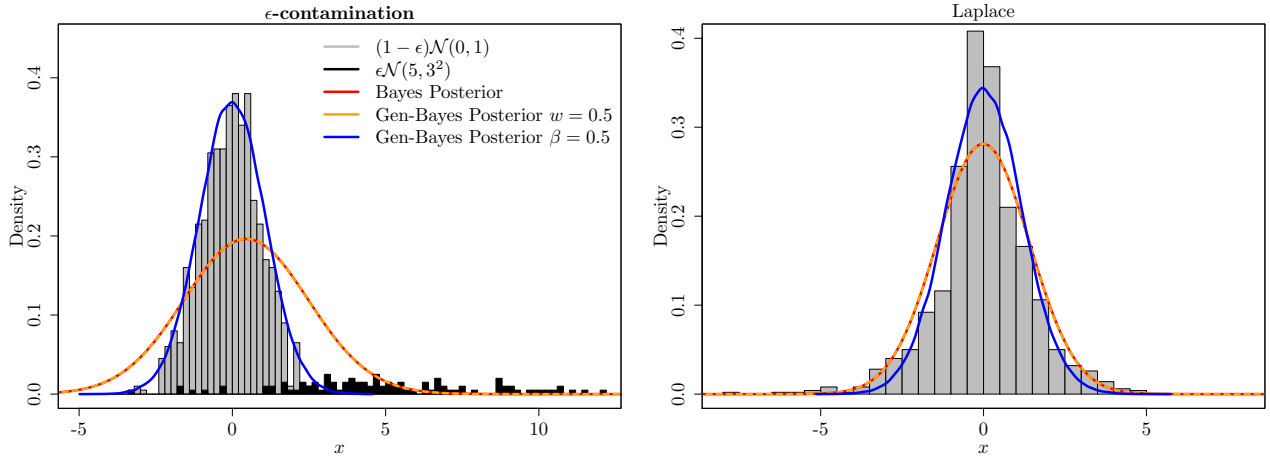

Figure A.2: General Bayesian Predictive densities from $\ell_0(x, f_\theta)$ (**red**), $\ell_w(x, f_\theta)$ (**orange**) and $\ell^{(\beta)}(x, f_\theta)$ (**blue**) given $n = 1000$ observations from $g_1(x) = 0.9\mathcal{N}(0, 1^2) + 0.1\mathcal{N}(5, 3^2)$ (**Left**) and $g_2(x) = \text{Laplace}(0, 1)$ (**Right**) under model $f_\theta = \mathcal{N}(\mu, \sigma^2)$

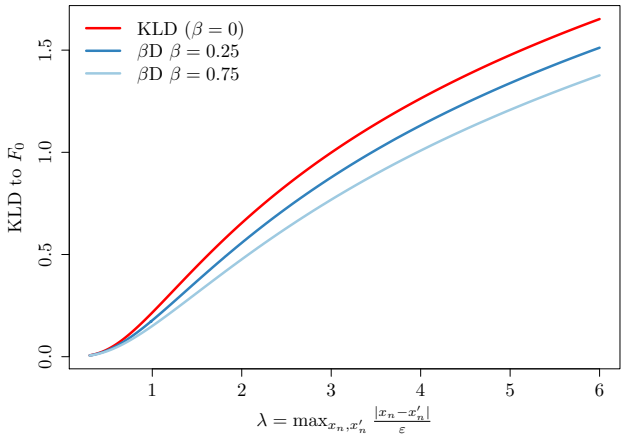

Figure A.3: KLD to true DGP F_0 of limiting inference of synthetic data generated by the Laplace mechanism with parameter $\lambda = \max_{x_n, x'_n} \frac{|x_n - x'_n|}{\varepsilon}$ using traditional Bayesian updating and βD minimising general Bayesian updating. The βD general Bayesian inference is able to get closer to F_0 than the traditional Bayesian inference uniformly for different privacy levels (parametrised by λ).

particular, Figure A.3 shows that for $f(\cdot;\theta) = \mathcal{N}(\cdot;\mu,\sigma^2)$ and $\mathcal{G}_{\varepsilon,\delta}$ as the Laplace mechanism with parameter $\lambda = \max_{x_n, x'_n} \frac{|x_n - x'_n|}{\varepsilon}$ (given by Proposition 3)

$$\text{KLD}\left(F_0 \,\|\, f_{\theta^{\beta\text{D}}_{\mathcal{G}_{\varepsilon,\delta}}}\right) < \text{KLD}\left(F_0 \,\|\, f_{\theta^{\text{KLD}}_{\mathcal{G}_{\varepsilon,\delta}}}\right). \tag{A.6}$$

for $\beta = 0.25$ and 0.75 uniformly for a range of values for λ.

A.2 Proof of Propositions

A.2.1 Proof of Proposition 1

Next we prove that for a given S-DGP $\mathcal{G}_{\varepsilon,\delta}$, model $f(\cdot;\theta)$ and infinite synthetic data sample $z_{1:\infty} \sim \mathcal{G}_{\varepsilon,\delta}$, there exists prior $\pi(\theta)$ and private DGP F_0 such that the L is able to get closer to F_0 in terms of D, than if they were to use the KLD limiting approximation to S-DGP $\mathcal{G}_{\varepsilon,\delta}$ $\theta^{\text{KLD}}_{\mathcal{G}_{\varepsilon,\delta}}$

Proposition 1 (Suboptimality of learning from the S-DGP)**.** *For* S-DGP $\mathcal{G}_{\varepsilon,\delta}$*, model* $f_\theta(\cdot)$*, and divergence* D*, there exists prior* $\tilde{\pi}(\theta)$*, private* DGP F_0 *and* $0 \leq m < \infty$ *such that*

$$T_{\ell_0}(m; D, f_0, g_{\varepsilon,\delta}) \leq D\left(F_0 \,\|\, f_{\theta^{\text{KLD}}_{\mathcal{G}_{\varepsilon,\delta}}}\right) \tag{A.7}$$

where $\theta^{\text{KLD}}_{\mathcal{G}_{\varepsilon,\delta}} := argmin_{\theta\in\Theta}\text{KLD}(\mathcal{G}_{\varepsilon,\delta} \,\|\, f_\theta)$ *and* T_{ℓ_0} *is the learning trajectory of the Bayesian posterior predictive distribution (using* ℓ_0*) based on (synthetic) data* $z_{1:m}$*,*

$$p(x \mid z_{1:m}) = \int f_\theta(x)\pi(\theta \mid z_{1:m})d\theta \tag{A.8}$$

.

$T_{\ell_0}(m; D, f_0, g_{\varepsilon,\delta}) = \mathbb{E}_z\left[D\left(F_0 \,\|\, p(\cdot \mid z_{1:m})\right)\right]$

Proof. Firstly fix the the divergence D between the KLD minimising model to $\mathcal{G}_{\varepsilon,\delta}$ and DGP F_0 as

$$K_\infty = D\left(F_0 \,\|\, f_{\theta^{\text{KLD}}_{\mathcal{G}_{\varepsilon,\delta}}}\right). \tag{A.9}$$

Now either $\min_{\theta\in\Theta} D(F_0 \,\|\, f_\theta)) = K_\infty$ also, in which case the D-minimising approximation to F_0 is the same distance from F_0 (in terms of distance D), as the KLD minimising approximation to $\mathcal{G}_{\varepsilon,\delta}$ and Eq. (A.7) hold with equality. Such a situation would happen if $\mathcal{G}_{\varepsilon,\delta} = F_0 = f(\cdot;\theta_0)$ for example. Or we can find π such that

$$T_{\ell_0}\left(m; D, f_0, g_{\varepsilon,\delta}\right) = \mathbb{E}_z\left[D\left(F_0 \,\|\, p(\cdot \mid z_{1:m})\right)\right] < K_\infty, \tag{A.10}$$

for example $\pi(\theta) = 1_{\theta'}$ for θ' such that $D(F_0 \,\|\, f(\cdot;\theta)) < K_\infty$ and therefore Eq. (A.7) holds with $m = 0$. □

We know that under regularity conditions and as $m \to \infty$, Bayes rule will concentrate about the parameter $\theta^{\text{KLD}}_{\mathcal{G}_{\varepsilon,\delta}} := \text{argmin}_{\theta\in\Theta}\text{KLD}(\mathcal{G}_{\varepsilon,\delta} \,\|\, f(\cdot;\theta))$ (Berk et al., 1966); as such we can conclude that given an infinite sample from an S-DGP $\mathcal{G}_{\varepsilon,\delta} \neq F_0$, it is not necessarily optimal to use all of the data available, contrary to the logic of standard statistical analyses.

Note that in general there is nothing about Proposition 1 that is specific about using traditional Bayesian updating and learning about $\theta^{\text{KLD}}_{\mathcal{G}_{\varepsilon,\delta}}$ in the limit. The proof of the proposition is unchanged if we consider for example general Bayesian updating using $\ell^{(\beta)}(x, f_\theta)$ and limiting parameter $\theta^{\beta\text{D}}_{\mathcal{G}_{\varepsilon,\delta}} := \text{argmin}_{\theta\in\Theta}\beta\text{D}(\mathcal{G}_{\varepsilon,\delta} \,\|\, f(\cdot;\theta))$.

A.2.2 Proof of Proposition 2

Next we provide a result that ensures we do not waste synthetic data when $\hat{m}$ is less than the maximum amount of synthetic data available. If more synthetic data is available than the $\hat{m}$ that are used for inference (e.g. sampling $z_{1:m}, m \to \infty$ from a GAN), we can average the posterior predictive distribution across different realisations and improve the performance of the predictive distribution if we consider convex proper scoring rules such as the logarithmic score. The proof of this result is simple and relies on Jensen's inequality.

Proposition 2 (Predictive Averaging)**.** *Given divergence D with convex scoring rule, s, averaging over different realisations (formulated using different realisations of $z_{1:m}$ indicated by superscript (b)) of the posterior predictive depending on different synthetic data sets improves inference:*

$$\mathbb{E}_z D\left(F_0 \,\|\, \frac{1}{B}\sum_{b=1}^{B} \tilde{p}\left(x \mid z^{(b)}_{1:m}\right)\right) \leq \mathbb{E}_z D\left(F_0 \,\|\, \tilde{p}\left(x \mid z^{(b)}_{1:m}\right)\right).$$

Proof.

$$\begin{aligned}
\mathbb{E}_z D\left(F_0 \,\|\, \frac{1}{B}\sum_{b=1}^{B} \tilde{p}\left(x \mid z^{(b)}_{1:m}\right)\right) &= \mathbb{E}_z\mathbb{E}_{x\sim f_0}\left[s\left(x, \frac{1}{B}\sum_{b=1}^{B}\tilde{p}\left(\cdot \mid z^{(b)}_{1:m}\right)\right)\right] - \mathbb{E}_z\mathbb{E}_{x\sim f_0}\left[s\left(x, f_0\right)\right] \\
&\leq \mathbb{E}_z\mathbb{E}_{x\sim f_0}\left[\frac{1}{B}\sum_{b=1}^{B} s\left(x, \tilde{p}\left(\cdot \mid z^{(b)}_{1:m}\right)\right)\right] - \mathbb{E}_z\mathbb{E}_{x\sim f_0}\left[s\left(x, f_0\right)\right] \qquad \text{(A.11)} \\
&= \frac{1}{B}\sum_{b=1}^{B}\mathbb{E}_z\mathbb{E}_{x\sim f_0}\left[s\left(x, \tilde{p}\left(\cdot \mid z^{(b)}_{1:m}\right)\right)\right] - \mathbb{E}_z\mathbb{E}_{x\sim f_0}\left[s\left(x, f_0\right)\right] \qquad \text{(A.12)} \\
&= \mathbb{E}_z\mathbb{E}_{x\sim f_0}\left[s\left(x, \tilde{p}\left(\cdot \mid z_{1:\hat{m}}\right)\right)\right] - \mathbb{E}_z\mathbb{E}_{x\sim f_0}\left[s\left(x, f_0\right)\right] \\
&= \mathbb{E}_z D\left(F_0 \,\|\, \tilde{p}\left(x \mid z^{(b)}_{1:m}\right)\right)
\end{aligned}$$

Where A.11 uses Jensen's inequality and the convexity of s and A.12 uses the identical distribution of $\tilde{p}\left(x \mid z^{(b)}_{1:m}\right)$ across varying b.

□

The significance of this is that more synthetic data can always be used to improve the predictive distribution, but not by naïvely using all of it to learn at once.

A.2.3 Elaboration of Remark 1

Another way to obtain $\hat{m}$ could be dependent on the concrete data stream through the consideration of the trajectory of the evaluation criteria for a concrete sequence of data items. This involves finding the minimum of:

$$\hat{m} := \text{argmin}_{0\leq m\leq M} \frac{1}{N}\sum_{j=1}^{N} s(x'_j, p^{\ell}(\cdot|z_{1:m}))$$

Finding this minimum in practice involves potentially adapting the upper bound via optimisation i.e. successively extending a search interval until the local minima is contained (see e.g. `http://www.optimization-online.org/DB_FILE/2007/10/1801.pdf`). This estimator depends on the order of the data; the performance of the resulting predictor can be improved using an analog of Proposition 2 by averaging across different shuffles.

A.3 Differential Privacy for Synthetic Data generated under the Normal-Laplace mechanism

Here we formalise how the Laplace mechanism (Dwork et al., 2014) provides synthetic data with differential privacy guarantees.

Proposition 3 (Synthetic Data via the Laplace Mechanism)**.** *Given real data $x_{1:n} \in D^n$, synthetic data, $z_{1:n} \in D^n$, generated according to the Laplace mechanism, $\mathcal{T}_{\mathcal{A}} : \mathbb{R}^n \to \mathbb{R}^n$ with $z_{1:n} = \mathcal{T}_{\mathcal{A}}(x_{1:n}) = x_{1:n} + \delta_{1:n}$ where $\delta_{1:n} \overset{i.i.d.}{\sim}$ Laplace$(0, \lambda)$, is $(\varepsilon, 0)$-differentially private with $\varepsilon = \max_{x_n, x'_n} \frac{|x_n - x'_n|}{\lambda}$.*

Proof. Fix $x_{1:n} \in D^n$ and consider $x'_{1:n} = \{x_{1:n-1}, x'_n\} \in D^n$

$$\begin{aligned}
&\left|\ln\left(\frac{P(\mathcal{T}_{\mathcal{A}}(x_{1:n}) = z_{1:n})}{P(\mathcal{T}_{\mathcal{A}}(x'_{1:n}) = z_{1:n})}\right)\right| \\
=&\left|\ln\left(\frac{P(x_{1:n} + \delta_{1:n} = z_{1:n})}{P(x'_{1:n} + \delta_{1:n} = z_{1:n})}\right)\right| \\
=&\left|\ln\left(\frac{\frac{1}{(2\lambda)^n}\exp\left(\frac{\sum_{i=1}^{n}|x_i - z_i|}{\lambda}\right)}{\frac{1}{(2\lambda)^n}\exp\left(\frac{\sum_{i=1}^{n}|x'_i - z_i|}{\lambda}\right)}\right)\right| \\
=&\left|\frac{|x_n - z_n|}{\lambda} - \frac{|x'_n - z_n|}{\lambda}\right| \\
\leq&\frac{|x_n - x'_n|}{\lambda},
\end{aligned}$$

As a result, such a procedure provides differential privacy of $\varepsilon = \max_{x_n, x'_n} \frac{|x_n - x'_n|}{\lambda}$ by Definition 1 of Dwork et al. (2006) □

In situations where $|x_n - x'_n|$ is unbounded, an artificial upper bound B can be imposed with corresponding truncation bounds $\{a, b\}$ with $b - a = B$. For example, this could take the form $\{a, b\} = \{\mu + \frac{B}{2}, \mu - \frac{B}{2}\}$ where μ is the mean of F_0. Any observation $x_i \notin (a, b)$ is instead considered as $\tilde{x}_i = \arg\min_{y\in\{a,b\}} |y - x_i|$ before the addition of the Laplace noise δ_i. Unlike Bernstein and Sheldon (2018) we cannot simply redact observations outside the truncation bounds as this would change the dimension of the response and leak privacy. As a result, the trunacated Laplace mechanism for unbounded real data is defined as $z_{1:n} = \mathcal{T}_{\mathcal{A}}(x_{1:n}) = \{\min(\max(a, x), b)\}_{1:n} + \delta_{1:n}$ with $\delta_{1:n} \overset{\text{i.i.d.}}{\sim}$ Laplace$(0, \lambda)$ and provides $(\varepsilon, 0)$-differential privacy with $\varepsilon = \frac{b-a}{\lambda}$.

We note here that the Laplace mechanism provides a more naïve and much simpler method for producing synthetic data compared with methods such as the DP-GAN (Xie et al., 2018) or PATE-GAN Jordon et al. (2018), and that clearly if estimation of variance is important then this mechanism constitutes a poor method to produce synthetic data. However, if one is interested in measures of central tendency, for example estimating coefficients for a

regression model then the Laplace mechanism will preserve such features in expectation across the data, which is not necessarily guaranteed by the GAN based methods. This shows that different methods, producing the same differential privacy guarantee, can have differing desirability to Learner, L, depending on which aspects of the DGP L wishes to capture.

A.4 Motivating Schematic

Figure A.4 provides an illustration representing our interpretation of the learning trajectory in the space of distributions for data. Such an interpretation is substantiated by the experiments in Section 4 of the main paper and further in Section A.6.5. We consider two cases, one where the synthetic data is able to get inference closer to F_0 and one where synthetic data immediately makes things worse under traditional Bayesian updating because the beliefs prior to observing the synthetic data were sufficiently informative. Note that 'distances' on this schematic are according to the chosen divergence D.

The learner L starts at their prior predictive before observing any data, $p(x) = \int f_\theta(x)\pi(\theta)d\theta$, and uses their own data $x^L_{1:n_L}$ and Bayes rule (Eq. (6) from the main paper and $\ell_0(z, f_\theta)$) to update their beliefs. Data $x^L_{1:n_L} \sim F_0$ is not privatised and thus using standard updating draws inference towards the DGP, F_0. By Bayesian additivity this posterior, $\pi(\theta \mid x^L_{1:n_L})$ given observations from F_0 becomes the prior for inference using observations $z_{1:m}$. We thus interpret any inference that L can do on their own data as providing a strongly informative prior about F_0. Again, we stress that our framework allows for the possibility that $n_L = 0$ here. However, we show later that $n_L > 0$ offers an inferential 'momentum' in the direction of F_0 under the βD and allows for the synthetic data to bring inferences closer to F_0 on the learning trajectory than is possible under traditional Bayesian updating.

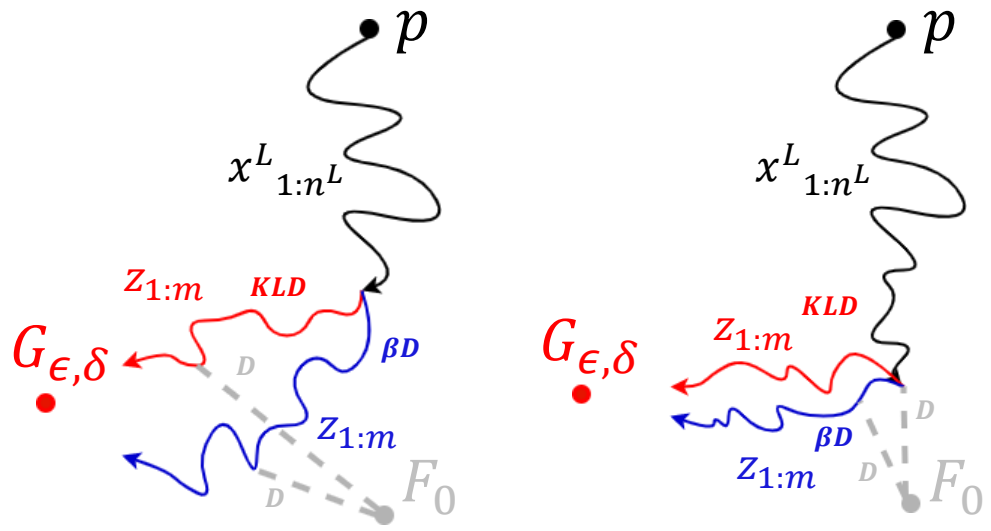

Figure A.4: The statistical geometry of learning using synthetic data: Starting from prior predictive (p), updating using $x^L_{1:n_L} \sim F_0$ takes inference towards F_0, before $z_{1:m} \sim \mathcal{G}_{\varepsilon,\delta}$ takes inference towards $\mathcal{G}_{\varepsilon,\delta}$ as $m \to \infty$. **Red** is traditional Bayesian updating (using $\ell_0(z, f_\theta)$) and **blue** is general Bayesian updating using $\ell^{(\beta)}(z, f_\theta)$. Distances are defined by the divergence D. Left: Fewer n_L mean that both learning methods are able to use synthetic data to improve inference for F_0 according to D. Right: Greater n_L means that before using synthetic data L is closer to F_0; when adding synthetic data here traditional Bayesian updating immediately takes inference away from F_0 with respect to D.

After the initial steps towards F_0 following the use of $x^L_{1:n_L}$, L starts learning from the synthetic data $z_{1:m}$ and therefore inference begins to move towards $\mathcal{G}_{\varepsilon,\delta}$. We argue that this does not imply inference is necessarily getting farther from F_0 providing acceptance of our minimal assumption that $\mathcal{G}_{\varepsilon,\delta}$ captures some useful information about F_0, i.e. we assume that in the model space defined by D, $\mathcal{G}_{\varepsilon,\delta}$ and F_0 are proximal. Such a phenomenon is depicted on the left-hand side of Fig. A.4: the red line corresponds to using Bayes' rule when updating using the synthetic data, and we indicate a point on this trajectory that is closer to F_0 (according to the chosen divergence D) than the inference using only $x_{1:n}$. Up until this point learning about $\mathcal{G}_{\varepsilon,\delta}$ was also helping to learn about F_0

but after such a point inference begins to be pulled away from F_0. However, on the right-hand side of Figure A.4 we acknowledge that this is not always the case, there can be situations where synthetic data immediately takes inference away from F_0. This will happen if the prior information (including the learner's own data) is very strong, or if the S-DGP is far from F_0.

Additionally, in blue we plot an example trajectory for learning using the βD. We argue that the βD has the ability to get closer to F_0 because of its robustness properties. The examples in Section A.1.1 have demonstrated that inference using the βD, unlike traditional Bayesian inference, is able to ignore outliers while still learning from inlying observations. As a result, initially the βD inference is able to learn from observations from the S-DGP that support the inference based on $x^L_{1:n_L}$ and dynamically downweight those that do not, therefore getting closer to F_0. Conversely, traditional Bayesian inference is influenced more by observations that disagree with F_0 and thus gets pulled more quickly towards $\mathcal{G}_{\varepsilon,\delta}$. This further reinforces the benefits that can be gained by beginning analysis using real data to impart some 'momentum' towards F_0 when learning from synthetic samples under the βD.

A.5 Prescribed Methodology

The discussion surrounding learning trajectories has demonstrated that a promising way to improve inference using synthetic data is to use the βD-loss combined with a reduced number of synthetic data items. A resulting question is how can one actually action such a procedure for inference given a realisation $z_{1:M} \sim \mathcal{G}_{\varepsilon,\delta}$, for $0 < M \leq \infty$ and independent testing set $x'_{1:N} \sim F_0$, for $0 < N \leq \infty$. To answer this we must address the following questions.

1. How exactly can realisations of real and synthetic datasets be used to estimate $\hat{m}$?
2. Given that we estimate $\hat{m} < M$, which $\hat{m}$ data items out of the M available should we use for inference?

We provide answers to these two questions in the next subsections.

A.5.1 Finding $\hat{m}$

The optimal m^* as defined in Eq. (12) of the main paper is an expectation over synthetic data $z \sim \mathcal{G}_{\varepsilon,\delta}$ and real data from the DGP, $x \sim F_0$, where the second expectation is hidden inside the definition of the Divergence (Definition 2). Given that, at best, we have a sample from the DGP, $x'_{1:N} \sim F_0$, and a sample from S-DGP $z_{1:M} \sim \mathcal{G}_{\varepsilon,\delta}$, m^* can be estimated by $\hat{m}$ as defined in Eq. (13) of the main paper. We note that even in the case where the S-DGP density could be given to the learner L, the expectation in Eq. (12) would likely be intractable and sampling would be required to estimate this integral regardless.

In the case where L is given the ability to sample from the S-DGP arbitrarily, they can continue to sample independently from $\mathcal{G}_{\varepsilon,\delta}$ to calculate $\hat{m}$. Clearly the more samples that they draw, the more accurate this estimation will be, but in reality fixing a computational sampling budget within this scheme is both inevitable and sensible.

When this is not the case, namely there exists $z_{1:M} \sim \mathcal{G}_{\varepsilon,\delta}$, for $0 < M < \infty$, the learner can repeatedly sample with or without replacement from $z_{1:M}$ to estimate $\hat{m}$. Clearly repeated sampling is only beneficial to the extent to which new samples are not too dependent on previous ones, something that will be determined by the relative size of the m's under consideration compared with M. If $\hat{m} \approx M$ (i.e. m is of the same order of magnitude as M) then estimating this integral from one sample is the best that the learner can do.

A.5.1.1 A p-value for the Use of Synthetic Data

In order to guard against the possible variance in calculating $\hat{m}$ from collections of real and synthetic data that might not be sufficiently large, we wish to provide a minimum guarantee that inference using $\hat{m}$ synthetic data samples is no worse than inference using only prior $\tilde{\pi}$ (including any of their own data and expert knowledge). In order to do so, the testing set $x'_{1:N} \sim F_0$ can be split into independent subsets, one of which is used to estimate $\hat{m}$ using Eq. (12) of the paper, whilst the other subset is used to construct a p-value that the divergence D is significantly reduced using $\hat{m}$ synthetic data samples compared to using no synthetic data at all, i.e. $m = 0$. The divergences are estimates as sums and therefore the central limit theorem can be invoked to construct such a p-value

Moreover, to guard against the variance in splitting a possible small testing set, this procedure can be repeated. For example, repeatedly splitting the data K times allows for the production of K p-values. Then a similar procedure to that considered in Watson and Holmes (2020) can be used to 'aggregate' these K p-values. They adapt a procedure of Meinshausen et al. (2009) which given a set of K p-values $\{p_1, \ldots, p_K\}$ produces valid aggregated p-value,

$$p^{(\text{median})}_{\text{aggregated}} := \min(1, \text{Median}(2p_1, \ldots, 2p_K)). \tag{A.13}$$

This facilitates a valid and robust test for whether to use synthetic data or not.

- If this test fails to reject the null hypothesis then the synthetic data from $\mathcal{G}_{\varepsilon,\delta}$ is disregarded ($\hat{m} = 0$) and the learner should just continue with $\tilde{\pi}$.
- If this test rejects the null hypothesis in favour of using synthetic data then $\hat{m}$ can be re-estimated using the whole testing set $x'_{1:N}$ and returned to the learner.

A.5.2 Inference Given $\hat{m}$

Given that L has estimated $\hat{m} < M$, how should they do inference using only $\hat{m}$ out of a possible M samples? Should they simply take the first $\hat{m}$ samples? This would introduce an undesirable dependence on the ordering of the data. Whilst this could be remedied by shuffling the data, we invoke Proposition 2 which shows that inference is improved by averaging posterior predictives over many synthetic data sets, $z_{1:\hat{m}}$ of size $\hat{m}$. By Proposition 2 the learner should repeatedly sample subsets of size $\hat{m}$ from the M available, with or without replacement, conduct posterior inferences using each one independently, and then average their posterior predictions. This has been shown to perform better in expectation according to divergence D than using any single synthetic data subset of size $\hat{m}$.

A.6 Additional Experimental Details and Results

This section details all of the information referred to in Section 4 of the main paper; it provides explicit model definitions, explicit grids that were explored to produce the plots included in the paper, further plots to these experiments, and other information regarding the reproduction of the code including reference to our GitHub repository etc.

A.6.1 Explicit Loss Function Formulations

In the two sections below, we formally define the models that were used in the experiments discussed in the main paper; the parameters that we searched across are formally introduced here as well, to be followed with a full experimental grid.

A.6.1.1 Simulated Gaussian

For the simulated Gaussian experiments, the first loss function is given by the standard log-likelihood for the posterior when learning the parameters of a Gaussian distribution, $f_\theta(x) = \mathcal{N}\left(x; \mu, \sigma^2\right)$. It can be written as below, with the addition of a w parameter to indicate that this loss function also encompasses the reweighting approach mentioned in Section 3.2:

$$\ell_w(x_i; \theta) = -w \cdot \log f(x_i; \theta) \tag{A.14}$$

Our second loss function leverages the βD in lieu of standard Bayesian updating (and its connection to the KLD) and can be written in closed form:

$$\ell_\beta(x_i, f(x_i; \theta)) = -w_\beta \left(\frac{1}{\beta} f(x_i; \theta)^\beta - \frac{1}{\beta + 1} \int f(z; \theta)^{\beta+1} dz \right) = -w_\beta \left(\frac{1}{\beta} f(x_i; \theta)^\beta - \left((2\pi)^{\frac{\beta}{2}} (1+\beta)^{\frac{3}{2}} \sigma^\beta \right)^{-1} \right), \tag{A.15}$$

where w_β is a 'calibration weight' (Bissiri et al., 2016), calculated via β and the data, that upweights the loss function to account for the 'cautiousness' of the βD. By this we mean that the βD-loss is generally smaller than the log-loss everywhere, rather than only in outlying regions of the data space.

In the case of these simulated examples we can also define a model (and associated log-loss function) that captures the noise-induced privatisation via the Laplace mechanism through the density of a Normal-Laplace convolution, which was first defined in (Reed, 2006), later corrected in (Amini and Rabbani, 2017) and now reformulated for our specific case of centred Laplace noise below:

$$\ell_{\text{Noise-Aware}}(x_i;\mu,\sigma,\lambda) = \\ -\log\left(\frac{1}{4\lambda}\left(e^{\frac{\mu-y}{\lambda}+\frac{\sigma^2}{2\lambda^2}}\left(1+\operatorname{erf}\left(\frac{y-\mu}{\sigma\sqrt{2}}-\frac{\sigma}{\lambda\sqrt{2}}\right)\right)+e^{\frac{y-\mu}{\lambda}+\frac{\sigma^2}{2\lambda^2}}\left(1-\operatorname{erf}\left(\frac{y-\mu}{\sigma\sqrt{2}}+\frac{\sigma}{\lambda\sqrt{2}}\right)\right)\right)\right) \tag{A.16}$$

A.6.1.2 Logistic Regression

In the case of our logistic regression examples on real-world datasets, w again allows us to reweight the standard log-likelihood for robustness when learning on the synthetic data to formulate our first loss function based on the logit-parameterised Bernoulli density function:

$$\ell_w(x_i;y_i,\alpha,\theta) = -w\cdot\log\Big(\text{logistic}\left(\alpha+x_i\cdot\theta\right)^{y_i}+\big(1-\text{logistic}\left(\alpha+x_i\cdot\theta\right)\big)^{(1-y_i)}\Big), \tag{A.17}$$

where:

$$\text{logistic}(x) = \frac{1}{1+e^{-x}}. \tag{A.18}$$

Applying the βD-loss to the same logit-parameterised Bernoulli density function becomes our second loss function:

$$\ell_\beta(x_i;y_i,\alpha,\theta) = -w_\beta\left(\frac{1}{\beta}\Big(\text{logistic}\left(\alpha+x_i\cdot\theta\right)^{y_i}+\big(1-\text{logistic}\left(\alpha+x_i\cdot\theta\right)\big)^{(1-y_i)}\Big)^{\beta}+ \right. \\ \left. \frac{1}{\beta+1}\Big(\text{logistic}\left(\alpha+x_i\cdot\theta\right)^{\beta+1}+\big(1-\text{logistic}\left(\alpha+x_i\cdot\theta\right)\big)^{\beta+1}\Big)\right) \tag{A.19}$$

Here, we cannot formulate a 'noise-aware' counterpart as the privatisation is via the black-box generations of the PATE-GAN.

A.6.2 Evaluation Criteria

Here we explicitly define each of our evaluation criteria, which are in general calculated via an evaluation set and an approximation to the posterior predictive using the samples drawn from MCMC chains. For these definitions, we let P and Q be two probability measures.

A.6.2.1 KLD

The KLD is defined as:

$$D_{\text{KL}}(P\parallel Q) = \int_{\mathcal{X}}\log\left(\frac{dP}{dQ}\right)dP$$

A.6.2.2 Log Score

The log score as in Gneiting and Raftery (2007) is a special case of a proper scoring rule defined as:

$$\mathbb{E}_{\theta\sim\pi(\theta|x_{1:n})}\left[\log f(z;\theta)\right] \tag{A.20}$$

A.6.2.3 Wasserstein Distance

Following Rueshendorff (1977) we define the Wasserstein distance as:

$$D_W(P,Q) = \sup\left(\int f dP-\int f dQ \mid \text{Lipschitz(f)}\leq 1\right).$$

Where $\text{Lipschitz}(f) = \sup_{x \neq y} \frac{|f(x)-f(y)|}{\|x-y\|}$

A.6.2.4 AUROC

For a probabilistic binary classification algorithm, the receiver operating characteristic (ROC) curve plots the true positive rate against the false positive rate as a parametric plot across different threshold settings. The area under this curve (AUROC) is equal to the probability that a classifier will rank some random positively labelled datapoint higher than a randomly chosen negatively labelled one; it comprises a common means of evaluating the performance of such classifiers.

$$A = \int_{x=0}^{1} \text{TPR}(\text{FPR}^{-1}(x))\, dx$$

where $\text{FPR}^{-1}(x)$ is the pseudo-inverse of the FPR that maps a false positive rate of x to the corresponding choice of threshold. Following Calders and Jaroszewicz (2007) it can also be estimated via:

$$\frac{\sum_{t_0 \in \mathcal{D}^0} \sum_{t_1 \in \mathcal{D}^1} \mathbf{1}[f(t_0) < f(t_1)]}{|\mathcal{D}^0| \cdot |\mathcal{D}^1|}$$

A.6.3 Reproducing Our Experiments

In order to reproduce the results shown in this supplement and in the main paper, one must execute large scale experiments to explore the effect of various parameters across large grids. This amounts to a significant computational workload that was facilitated by recent advances in probabilistic programming in Julia's Turing PPL (Ge et al., 2018), MLJ (Blaom et al., 2020) and Stan (Carpenter et al., 2017) and the use of large compute nodes. Specifically, we predominantly relied upon a SLURM cluster managed by the University of Warwick's RTP research computing platform.

All of the code, experimental configuration specifications and other requirements are laid out in our GitHub repository[1]; below are the parameter ranges and other values used to produce the plots in the case of the two experiment types discussed in the paper:

- 6000 MCMC samples were taken per chain in the case of logistic regression; 4000 in the case of the Gaussian simulations. In both cases, 500 warm-up samples were sampled and subsequently discarded.
- We used Stan's NUTS (Hoffman and Gelman, 2014) sampler and the NUTS sampler provided by Turing to carry out the majority of the inference tasks; we monitored $\hat{R}$ and other convergence criteria when designing the experiments and during their execution to ensure consistent convergence.
- For the βD based models, $\beta \in \{0.1, 0.2, 0.25, 0.4, 0.5, 0.6, 0.75, 0.8.0.9\}$. We also upweight each posterior using a multiplicative $w_\beta = 1.25$ to account for the fact that the βD is more 'cautious' in general than standard updating.
- For the standard reweighted models, $w \in \{0.0, 0.25, 0.5, 0.75, 1.0\}$.
- Data quantities:
  - For the simulated Gaussian experiment we jointly varied n and m with $\text{n} \in \{2, 4, 6, 8, 10, 13, 16, 19, 22, 25, 30, 35, 40, 50, 75, 100\}$ and $m \in \{1, 2, \ldots, 99, 100, 120, 140, 160, 180, 200\}$.
  - For logistic regression we traversed a grid of proportional quantities of data rather than explicit n's for ease of comparison across the chosen datasets, $\alpha_{\text{real}} \in \{0.025, 0.05, 0.1, 0.25, 0.5, 1.0\}$ and $\alpha_{\text{synth}} \in \{0.0, 0.01, 0.02, 0.03, 0.04, 0.05, 0.075, 0.1, 0.15, 0.2, 0.4, 0.6, 1.0\}$.

  Note that in both cases, we also ran a consecutive stream of real data values without any synthetic data to produce the black lines plotted on the branching plots and to give a means of comparing the performance of synthetic data, through the expected minima (or maxima in the case of AUROC), with the best case scenario of more real data allowing us to calculate the approximate effective number of real samples that the synthetic data could provide (see the following Section A.6.4).

[1] `https://github.com/HarrisonWilde/Foundations-of-Bayesian-Learning-from-Synthetic-Data`

- For evaluation, we generated an additional 500 samples from F_0 for the Gaussian, and utilised 5-fold cross validation for the logistic regressions where one fold was used for evaluation in each of the five steps.
- Priors:
  - In the case of our Gaussian model, we placed conjugate priors on θ with $\sigma^2 \sim \text{InverseGamma}(\alpha_p, \beta_p)$ and $\mu \sim \mathcal{N}(\mu_p, \sigma_p * \sigma)$. Here we set $\alpha_p = 2.0, \beta_p = 4.0, \mu_p \in \{0.0, 1.0, 3.0\}, \sigma_p \in \{1, 10, 30\}$.
  - We used uninformative Gaussian priors for the logistic regressions' α (the intercept) and θ (other parameters), e.g. $\alpha, \theta \sim \mathcal{N}(0, 50)$
- For the logistic regression models, we initialised α and θ through 3 varying approaches:
  1. Using MLJ's `LinearRegression` model to calculate the MLE given the step's amount of real data n.
  2. Setting θ to be a vector of 0's matching the dimension of the dataset
  3. Randomly initialising θ within a locality of 0 using a standard Gaussian model.

  It is worth noting that this initialisation was not seen to have much observable effect in terms of MCMC convergence; the number of samples were carefully chosen alongside effective sampling schemes such as NUTS and HMC to ensure convergence in almost all cases, even with very little or noisy data.
- We repeated all of the configurations defined by combinations of the parameter values specified above at least 100 times to ensure reasonable certainty in our results in the presence of multiple sources of noise (data generation, privatisation and MCMC). During each of these 'full iterations' we specified and recorded a randomised seed to ensure that the real data used was reshuffled or different each time for the logistic regression and simulated Gaussian experiments respectively. This then allowed us to calculate expected curves across different realisations of varying amounts of real data.
- We used the PATE-GAN configuration and implementation provided by the original paper's repository[2], but did alter the number of teachers to 100 in the case of the Framingham dataset as it was suggested this quantity should increase alongside the size of the training dataset. As mentioned above, experiments were repeated many times and with different holdout sets from the original data for evaluation.

A.6.3.1 Datasets Used for Logistic Regression

The Framingham Cohort Dataset contains 4240 rows and 15 predictors, a mix of binary labels and continuous or discrete numerics. The label is a binary indicator for someone's ten year risk of coronary heart disease. Many of the columns such as age, education, cigarettes smoked per day and more pose genuine privacy concerns to the subjects of this dataset.

The UCI Heart Dataset contains 303 rows and 14 predictors, again a mix of binary labels and continuous or discrete numerics. The label is a binary indicator for the presence of heart disease in a subject. Many of the attributes pose genuine privacy concerns to the subjects of this dataset.

A.6.4 Elaboration on the Formulation and Meaning of the Figures

The following section further details how each type of figure shown in the paper is made, and how they should be interpreted:

- **The 'branching' plots** (as in Figures A.7, A.8, A.9, A.10, A.11, A.12, A.13, A.14, A.15, A.16 and Figure 1 of the main paper (and Figure 4 which is a special case)) show the total amounts of data on the x axis used to train various model via MCMC, each model corresponding to a single point on the plot. This amount is totalled in the sense that it corresponds to some amount of real data n_L added on to some amount of synthetic data m. Each 'branch' of these plots fixes the amount of real data it represents at the root of its branch from the black line which represents a varying amount of real data and *no* synthetic samples. Each branch is then colour coded and corresponds to some fixed quantity of real data n_L plus a varying amount of

[2] `https://bitbucket.org/mvdschaar/mlforhealthlabpub/src/0b0190bcd38a76c405c805f1ca774971fcd85233/alg/pategan/`

synthetic data, such that the amount of synthetic samples included in learning up to a point on the x axis can be calculated by subtracting the fixed real data quantity n_L from the x axis value. The y axis is relatively clear in corresponding to the relevant criteria value when the model trained at each point of the branching curves is evaluated. Note that these plots directly show the 'learning trajectory' as depicted in Figure A.4

- **The model comparison plots** (as in Figures A.17, A.18, A.19, A.20, A.21, A.22, A.23, A.24, A.25, A.26 and Figure 3 of the main paper) are in some ways just a more specific view of the 'branching' plots discussed above, in that they fix the real amount of data to some n_L and simply illustrate the performance of the models under varying synthetic data quantity m alongside one another. This is essentially a layering of a single consistent branch from each 'branching' plot layered on top of each other across all of the model configurations of interest.

- **The n-effective plots** (as in Figures A.27, A.28, A.29 and Figure 2 of the main paper) illustrate the maximal effective number of real samples that can be gained through the use of synthetic data under varying real data quantity n_L. In order to illustrate this, we calculate bootstrapped (Efron and Tibshirani, 1994) mean and variance of the minima / maxima by first taking the expectation over each seed / iteration / realisation of a curve alongside the synthetic data varying; this is done separately for each 'branch' (i.e. fixed real data quantity n_L) to get an expected curve for each. The turning point of these curves is then matched with the closest realised point along the black line representing varying amounts of real data without any synthetic data in order to produce an estimate for the amount of real data samples this turning point effectively corresponds to. Namely, the additional number of real samples required to achieve the same minimum / maximum criteria value when learning using the optimal amount of synthetic data.

  This process is done in the bootstrap paradigm such that we repeatedly sample $N = 100$ seeds / iterations / curves from those collected during the full experiment, the turning point corresponding to this expectation over N curves is then calculated; this is repeated $B = 200$ times to then calculated a bootstrapped mean and variance for the best (i.e. at the turning point in synthetic data) effective number of real samples for each amount of fixed real data.

  This can be expressed as below, by taking an evaluation set $x'_{1:n'} \sim F_0$ and then by calculating B $n_{\text{eff}}^{(b)}$s for each real data quantity n_L that we take, alongside varying synthetic samples. Each $n_{\text{eff}}^{(b)}$ is a bootstrap n-effective sample formulated using the the minimum of the expected curve arising across a randomly sampled $N = 100$ seeds / iterations / curves that is then 'matched' with a similar expected curve arising from $R = 100$ sampled real-only (black lines) seeds / iterations / curves to provide an estimation for the number of real samples each minimum represents:

$$n_{\text{eff}}^{(b)} = \text{argmin}_t \left| \frac{1}{M} \sum_{j=1}^{M} s(x', p(\cdot \mid x^{(j)}_{1:n_L+t})) - \min_m \frac{1}{N} \sum_{i=1}^{N} s(x', \tilde{p}(\cdot \mid x^{(i)}_{1:n_L} z^{(i)}_{1:m})) \right|$$

  This gives rise to a collection of B bootstrap samples for each value of n_L from which we can compute a bootstrapped mean and variance to present in our n-effective plots.

A.6.5 Further Results and Figures

Figure A.5 shows the relationship between privacy and model misspecification in terms of the KLD. We observe that initially, using a small amount of synthetic data is preferable due to the amount of noise a small ε introduces; there is then a cross over point around $\varepsilon = 1$ to $\varepsilon = 10$ where using more data becomes more desirable as the level of noise decreases, until eventually at $\varepsilon = 100$ we see comparable performance from using the synthetic data to using the same amount of real data. This pattern of usefulness is slightly more complex in the case of a GAN based model as the usefulness of the data also relies on the convergence of the GAN and how representative its generated samples are through the effectiveness of training, regardless of the value of ε.

We can conduct a more fundamental investigation of the PATE-GAN's behaviour under varying ε by observing through Figure A.6 the average predictor standard deviation in the resulting datasets generated by the GAN under different ε specifications. This allows us to observe the 'mode collapse' of the generative model when ε is sufficiently small and privacy is sufficiently high. Interestingly, as observed in Figure 4 from the main paper, this data is still somewhat useful, at least in small quantities, in learning about F_0.

A.6.5.1 Branching Plots

Figures A.7, A.8, A.9, A.10, A.11, A.12, A.13, A.14, A.15, A.16 show the full suite of branching plots for all of our experiments. In each of these plots we investigate different privatisation levels and criteria of interest; we then draw comparisons amongst the model configurations. In particular we see the finite and asymptotic effectiveness of the βD across a wide range of β when compared to a range of standard and reweighted approaches. The 'Noise-Aware' models perform the best as is expected as these models are aware of the privatisation process and thus can go some way towards modelling it. In terms of KLD especially, we can see this asymptotic effectiveness via the black dashed line representing $\text{KLD}(F_0 \,\|\, f_{\theta^*})$ for $\theta^* = \theta^{\text{KLD}}_{\mathcal{G}_{\varepsilon,\delta}}$ in the case of models involving w and $\theta^* = \theta^{\beta\text{D}}_{\mathcal{G}_{\varepsilon,\delta}}$ for models involving β. These quantities represent the approximation to F_0 given an infinite sample from $\mathcal{G}_{\varepsilon,\delta}$ under the two model types. It can be seen that a relatively small increase in noise / privatisation in the Gaussian experiments can quite drastically change the effectiveness of using synthetic data, such that only when very little real data is available should its use be considered at all. For the logistic regression experiment datasets, we actually see a reduction in performance when using the βD on the UCI Heart dataset. This is likely due to the βD's natural tendency to downweight samples, as upon inspection it appears that the two model families will converge to roughly similar criterion values.

A.6.5.2 Model Comparison Plots

Figures A.17, A.18, A.19, A.20, A.21, A.22, A.23, A.24, A.25, A.26 show the full suite of model comparison plots for all of our experiments. These plots allow us to directly compare the performance of different model configurations given an identical amount of real data n_L and an accompanying, varying amount of synthetic data. We can compare the most desirable $\hat{m}$ values achieved by all of the models to observe that again, other than in the case of the UCI Heart dataset, the βD consistently performs well in comparison to other approaches. The models where w is set to 0 offer a baseline of sorts, with the points where each other model's curve crosses its straight line representing when the various other model configurations stop being justifiable and synthetic data should not be used at all. It can be seen that not only does the βD achieve more desirable performance in the majority of cases, but it also remains robust to large amounts of synthetic data where other approaches fail to be effective and quickly lose out through the use of synthetic data. We offer all of these plots on fixed axes across each grid so that a reader can notice the narrowing scope and magnitude to which synthetic data is useful as more real data samples become available; this highlights the advantages of the βD in that it is a 'safer' option even in the situations where synthetic data cannot help the inference much in that it is more robust to the damage it can cause, especially when a user may not be able to calculate $\hat{m}$ explicitly.

A.6.5.3 n-Effective Plots

Figures A.27, A.28, A.29 show the full suite of n-effective plots for all of our experiments, other than for the Framingham dataset which is already included in the main text. In these plots we observe some interesting phenomena, primarily in that the number of real effective samples to gain through synthetic data is related to the amount of real data that has already been used; in general we observe asymptotic behaviour in the criteria as the amount of real data increases meaning variation in the performance of synthetic data and the resulting turning points can indicate a greater amount of effective samples gained despite the actual criterion value improvement being marginal. As such, reading the x axis is in some ways misleading in that effective samples often 'mean more' in the sense that they indicate a greater improvement in the criteria under a smaller total amount of data. Again, we see that in the case of the UCI Heart dataset, the improvements offered by the βD are less significant and in some cases non-existent over standard approaches.

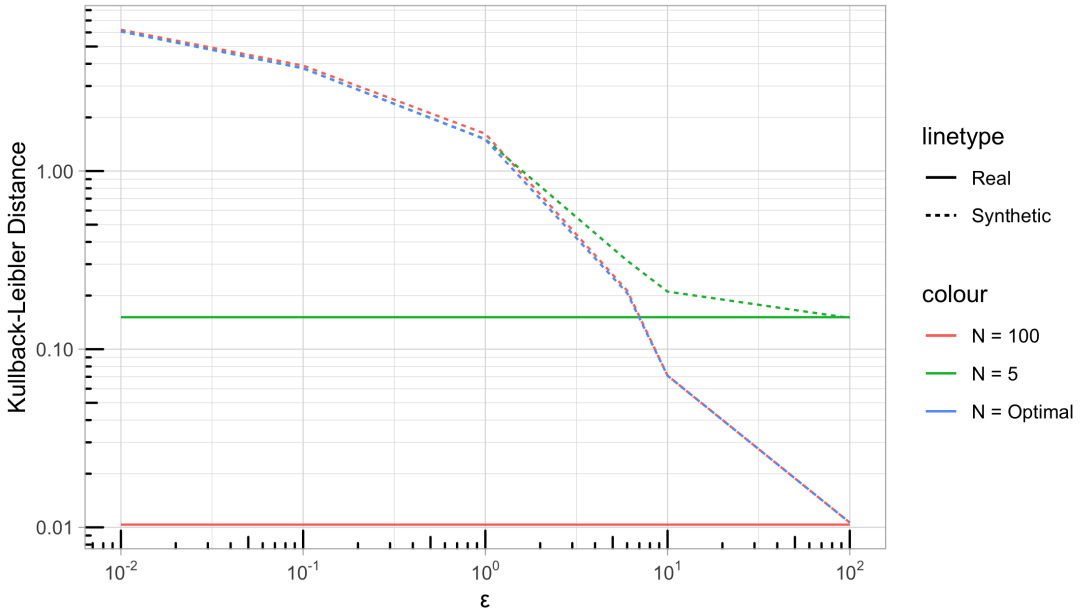

Figure A.5: This plot shows the average KLD between F_0 and the models arising from the specified amounts of real or synthetic data under varying DP-guarantees, the line types distinguish between real and synthetic, and colour indicates the data quantity used for learning. Here 'optimal' indicates that $\hat{m}$ samples are used under each ε. As $\varepsilon \to \infty$ the synthetic data becomes arbitrarily close to samples from F_0 through the Laplace mechanism such that the question of whether to use any or all of the synthetic data is most interesting at lower ε[3].

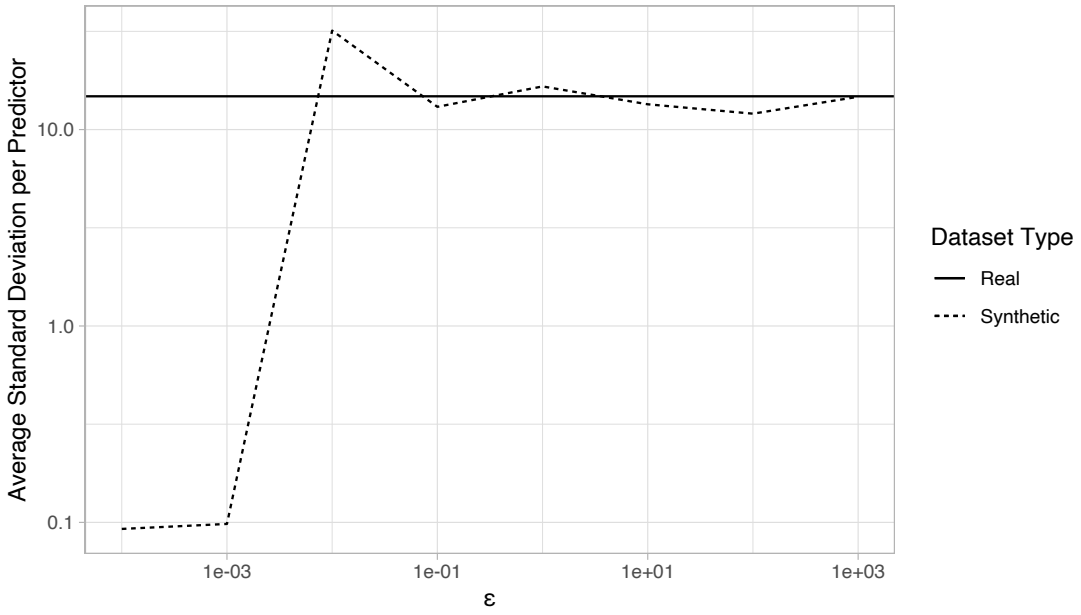

Figure A.6: Here, we present the averaged predictor standard deviation for datasets arising from various ε values. Similarly to Figure A.5, as $\varepsilon \to \infty$ the synthetic data should more closely resemble the real dataset that was used to train it, plus whatever complications arise by nature of this training. It can be seen that as privacy increases with $\varepsilon \to 0$ that there is a point $\varepsilon^* \in [10^{-3}, 10^{-2}]$ where the predictors' standard deviation collapses.

[3]Note that this is not consistently the case for GAN based methods as the utility of synthetic data is also limited by how well the GAN can initially capture F_0 from its training data regardless of the chosen ε.

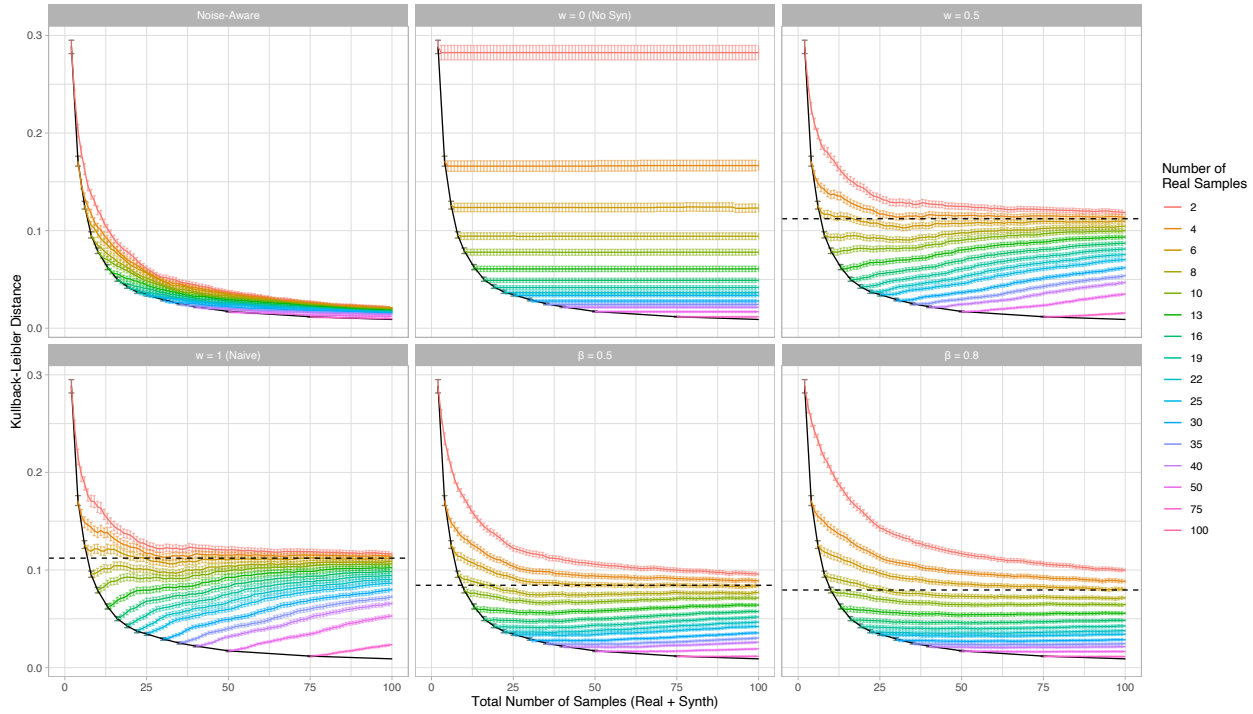

Figure A.7: Branching plots for each model configuration in the case of the simulated Gaussian experiments illustrating the KLD against the total number of samples where DP of $\varepsilon = 8$ is achieved by the Laplace mechanism via noise of scale $\lambda = 0.75$.

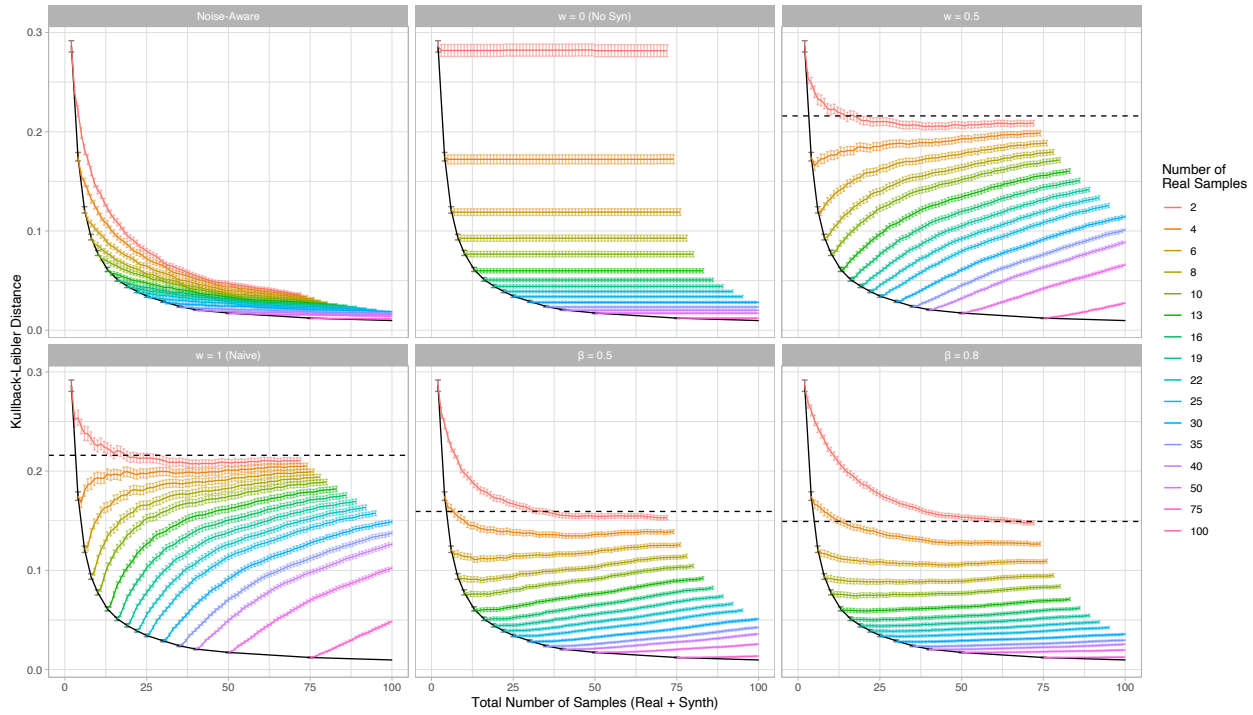

Figure A.8: Branching plots for each model configuration in the case of the simulated Gaussian experiments illustrating the KLD against the total number of samples where DP of $\varepsilon = 6$ is achieved by the Laplace mechanism via noise of scale $\lambda = 1.0$.

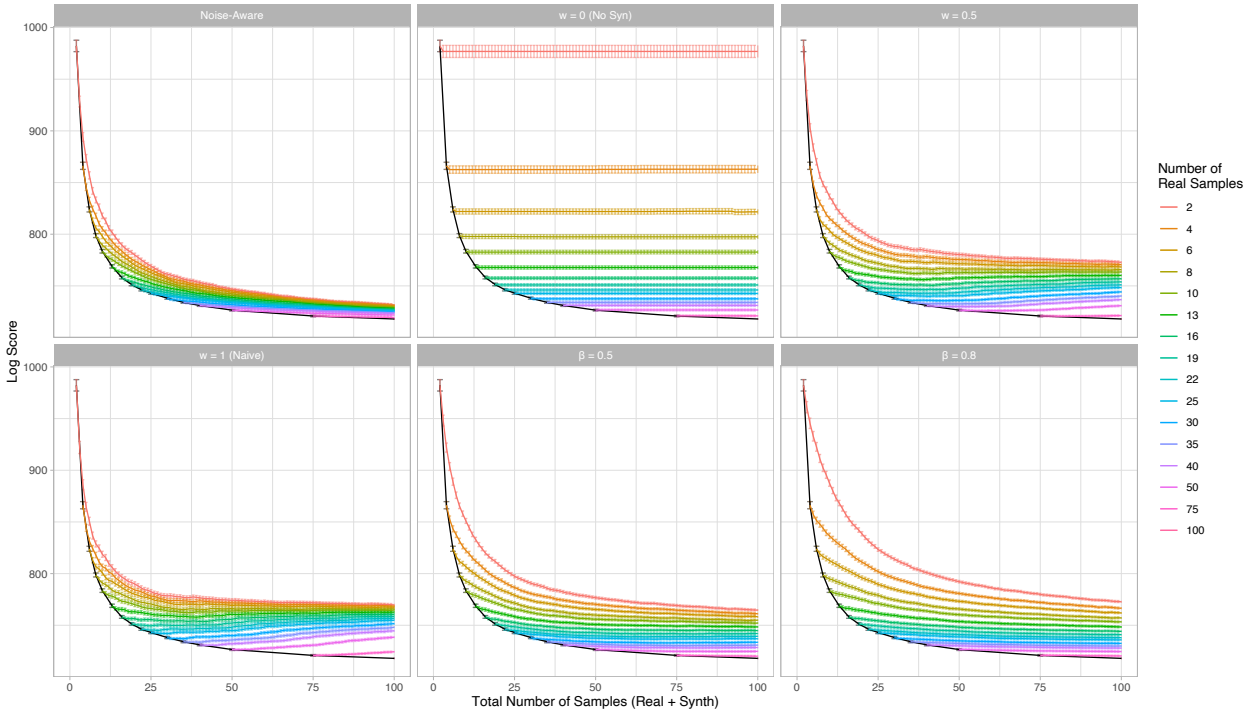

Figure A.9: Branching plots for each model configuration in the case of the simulated Gaussian experiments illustrating the log score against the total number of samples where DP of $\varepsilon = 8$ is achieved by the Laplace mechanism via noise of scale $\lambda = 0.75$.

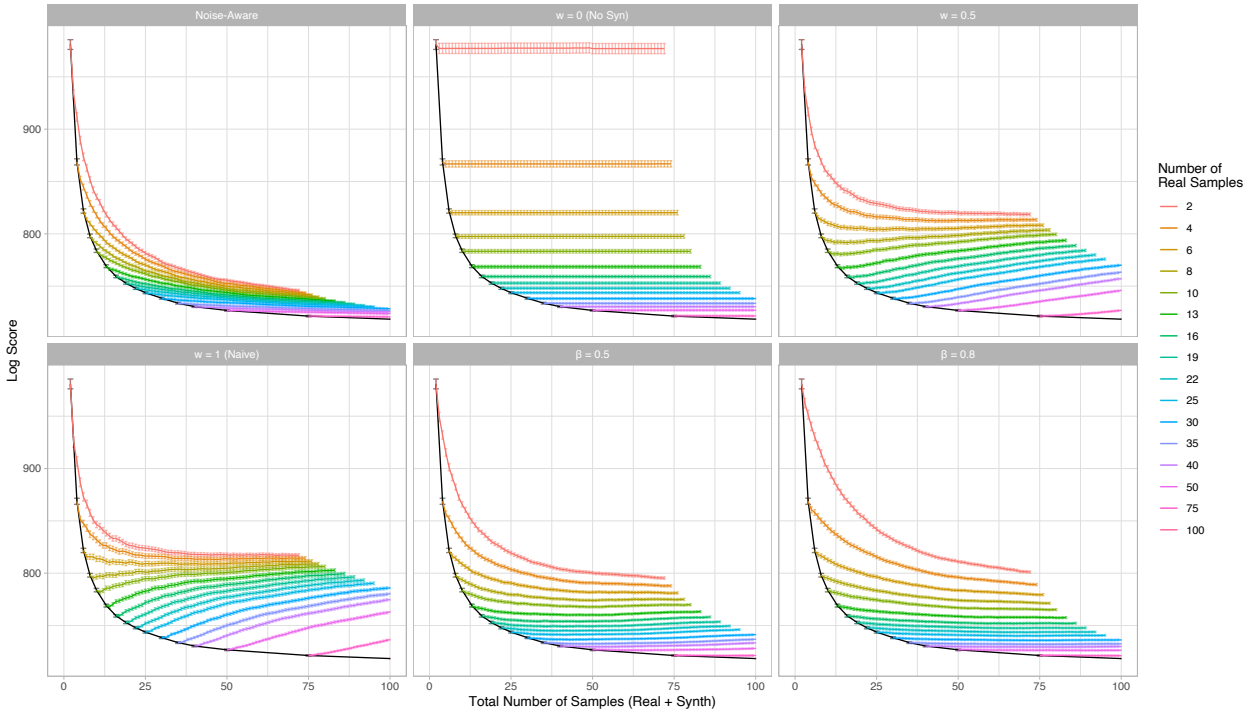

Figure A.10: Branching plots for each model configuration in the case of the simulated Gaussian experiments illustrating the log score against the total number of samples where DP of $\varepsilon = 6$ is achieved by the Laplace mechanism via noise of scale $\lambda = 1.0$.

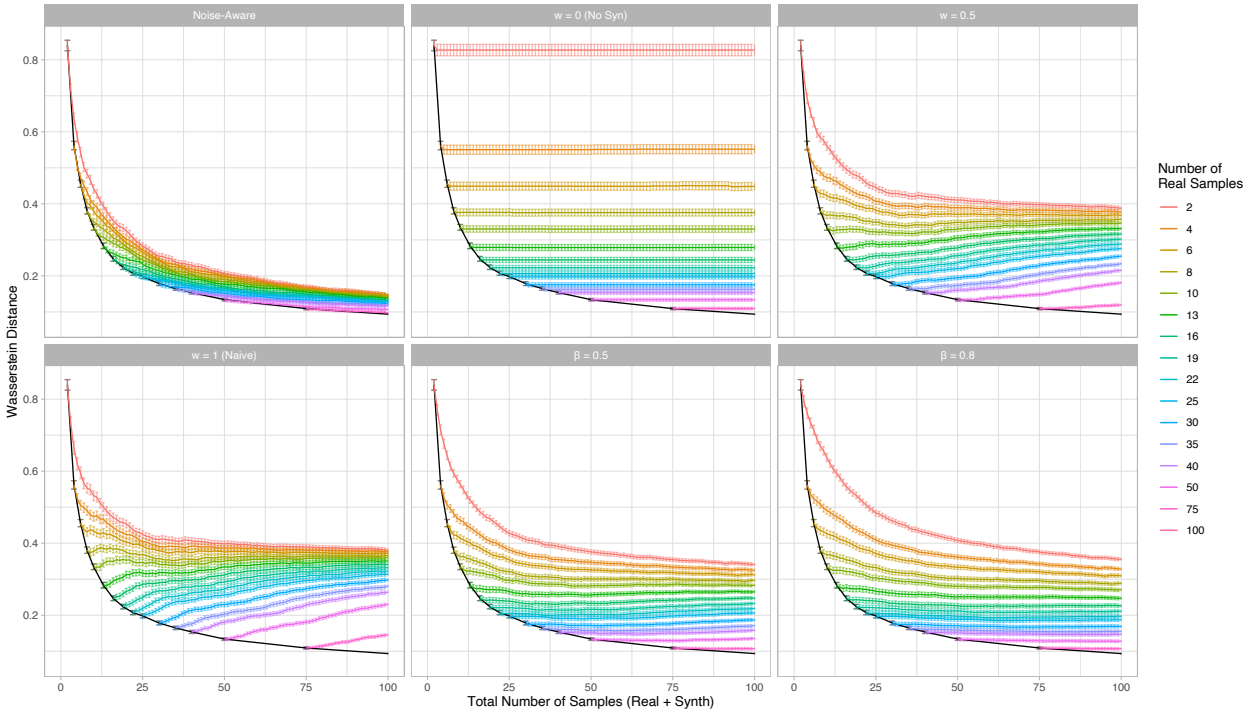

Figure A.11: Branching plots for each model configuration in the case of the simulated Gaussian experiments illustrating the Wasserstein distance against the total number of samples where DP of $\varepsilon = 8$ is achieved by the Laplace mechanism via noise of scale $\lambda = 0.75$.

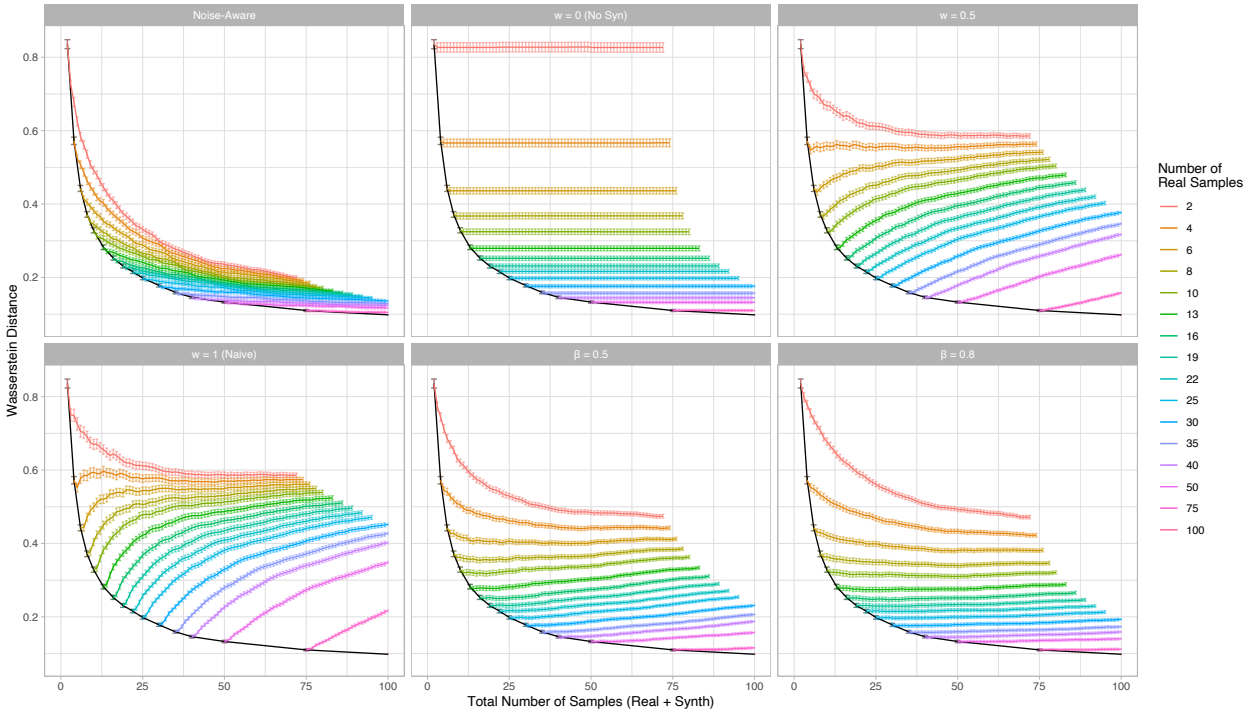

Figure A.12: Branching plots for each model configuration in the case of the simulated Gaussian experiments illustrating the Wasserstein distance against the total number of samples where DP of $\varepsilon = 6$ is achieved by the Laplace mechanism via noise of scale $\lambda = 1.0$.

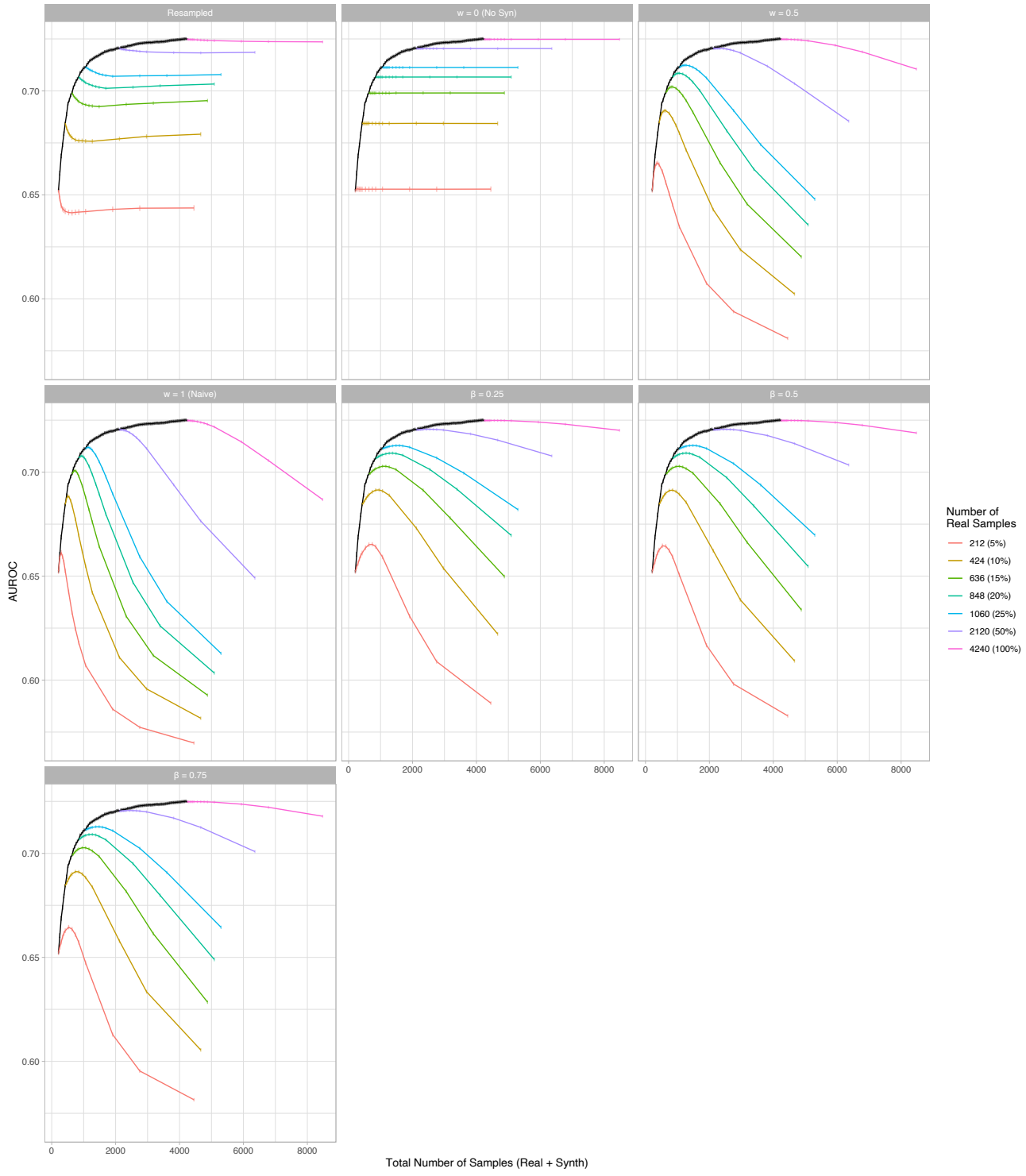

Figure A.13: Branching plots for each model configuration in the case of the logistic regression experiments on the Framingham dataset illustrating the AUROC against the total number of samples where DP of $\varepsilon = 6$ is achieved via generation of synthetic datasets using the PATE-GAN.

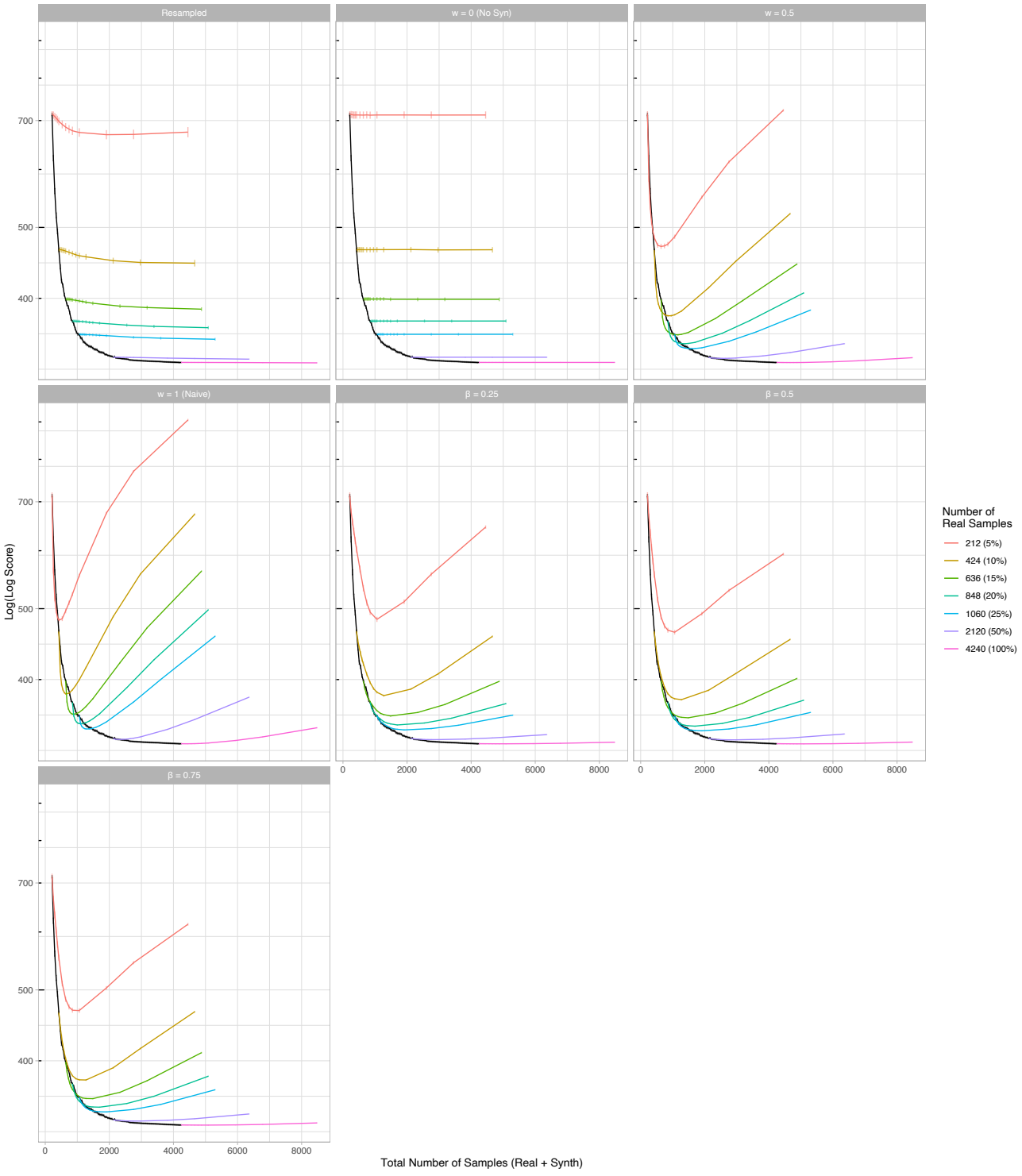

Figure A.14: Branching plots for each model configuration in the case of the logistic regression experiments on the Framingham dataset illustrating the log score against the total number of samples where DP of $\varepsilon = 6$ is achieved via generation of synthetic datasets using the PATE-GAN.

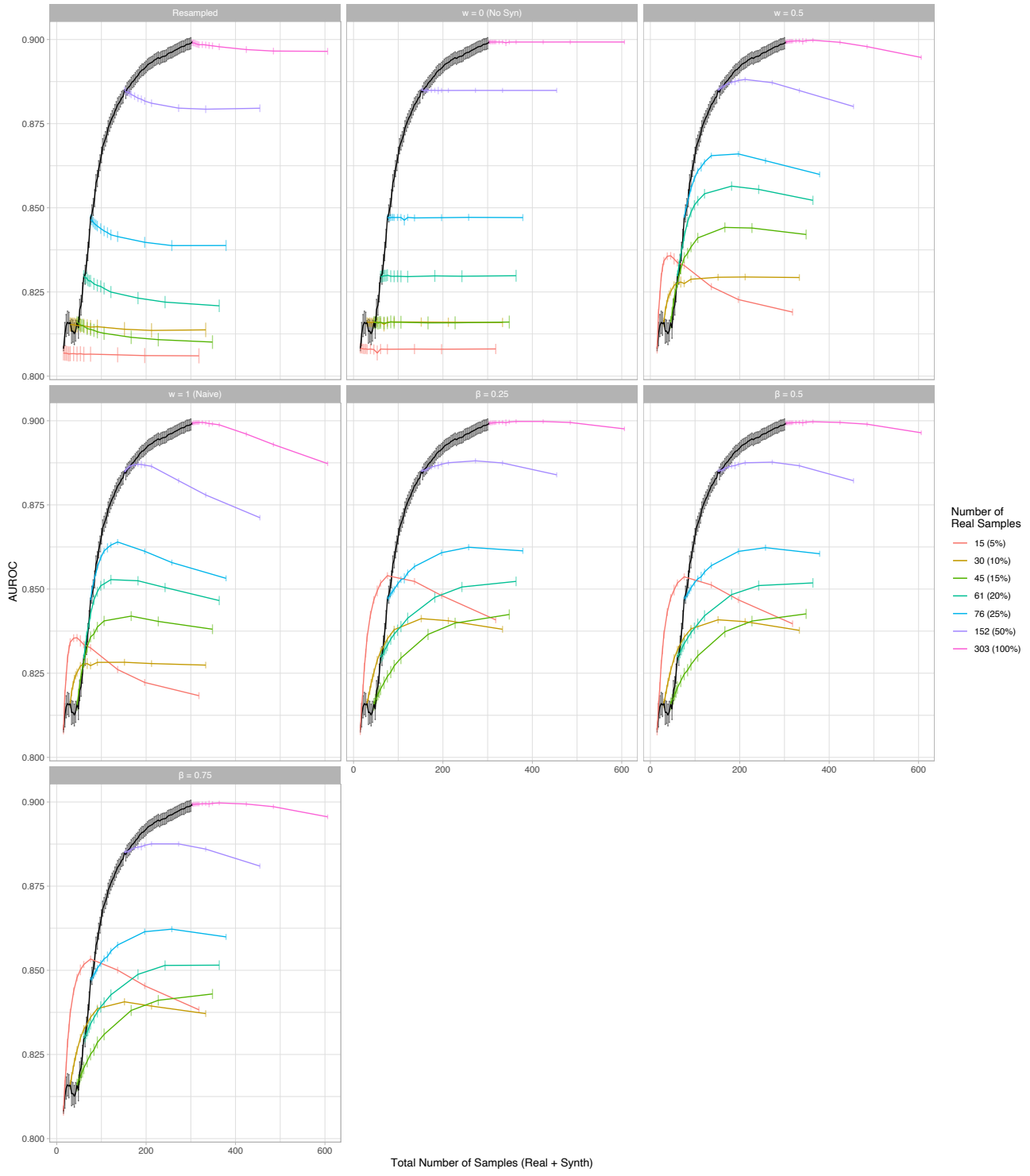

Figure A.15: Branching plots for each model configuration in the case of the logistic regression experiments on the UCI Heart dataset illustrating the AUROC against the total number of samples where DP of $\varepsilon = 6$ is achieved via generation of synthetic datasets using the PATE-GAN.

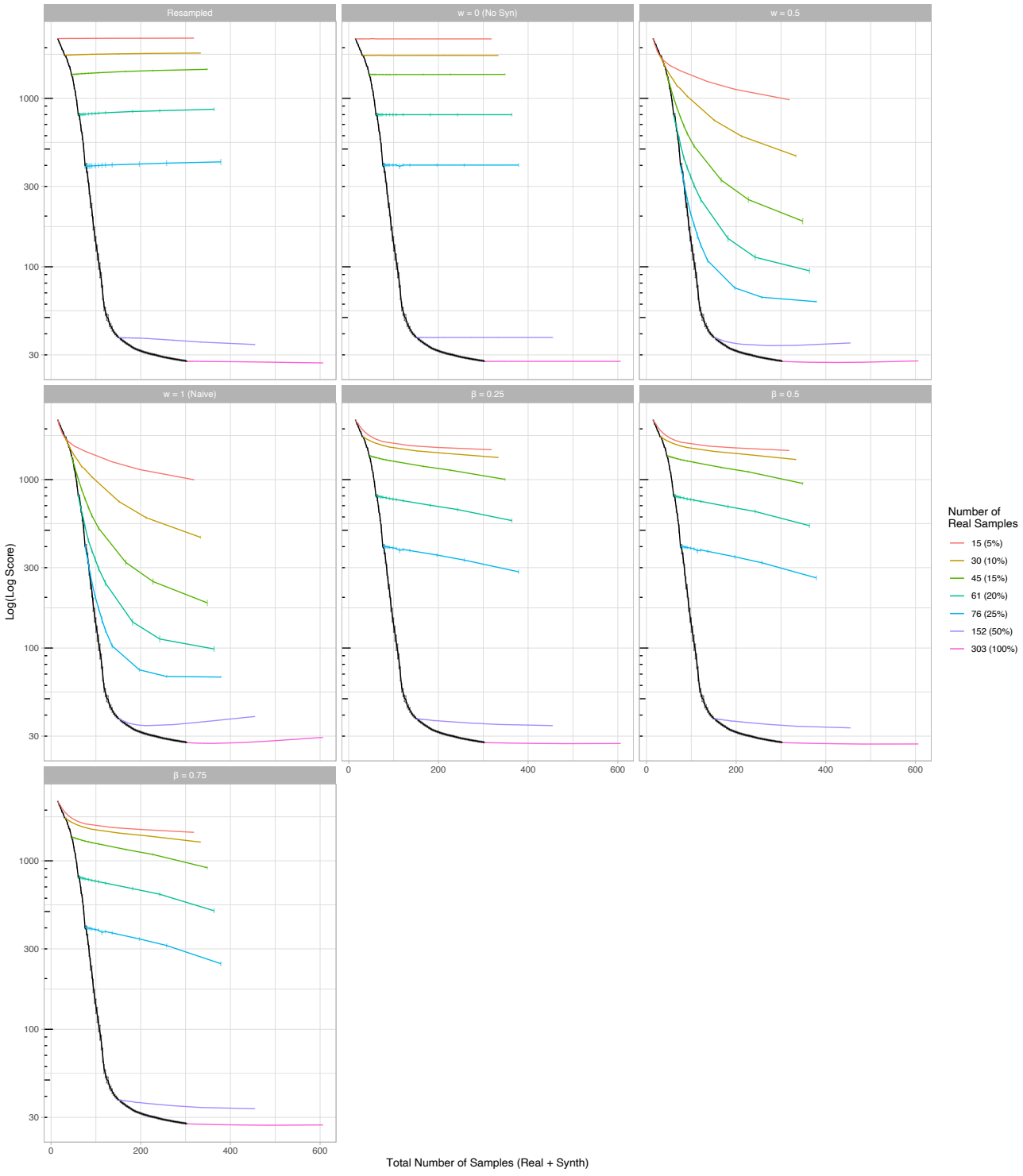

Figure A.16: Branching plots for each model configuration in the case of the logistic regression experiments on the UCI Heart dataset illustrating the log score against the total number of samples where DP of $\varepsilon = 6$ is achieved via generation of synthetic datasets using the PATE-GAN.

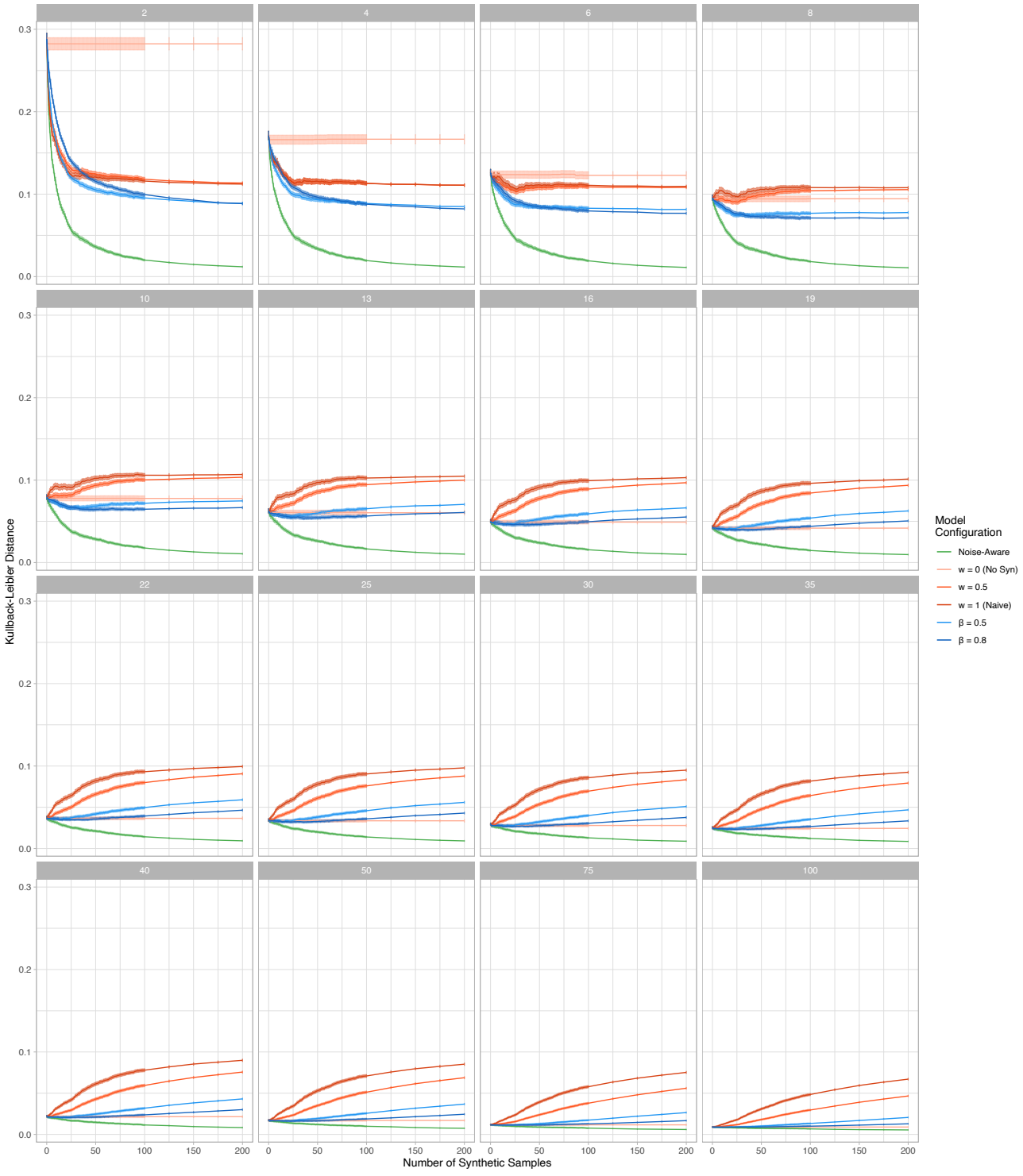

Figure A.17: Model comparison plots for each real data quantity n_L in the case of the simulated Gaussian experiments illustrating the KLD against the number of synthetic samples where DP of $\varepsilon = 8$ is achieved by the Laplace mechanism via noise of scale $\lambda = 0.75$.

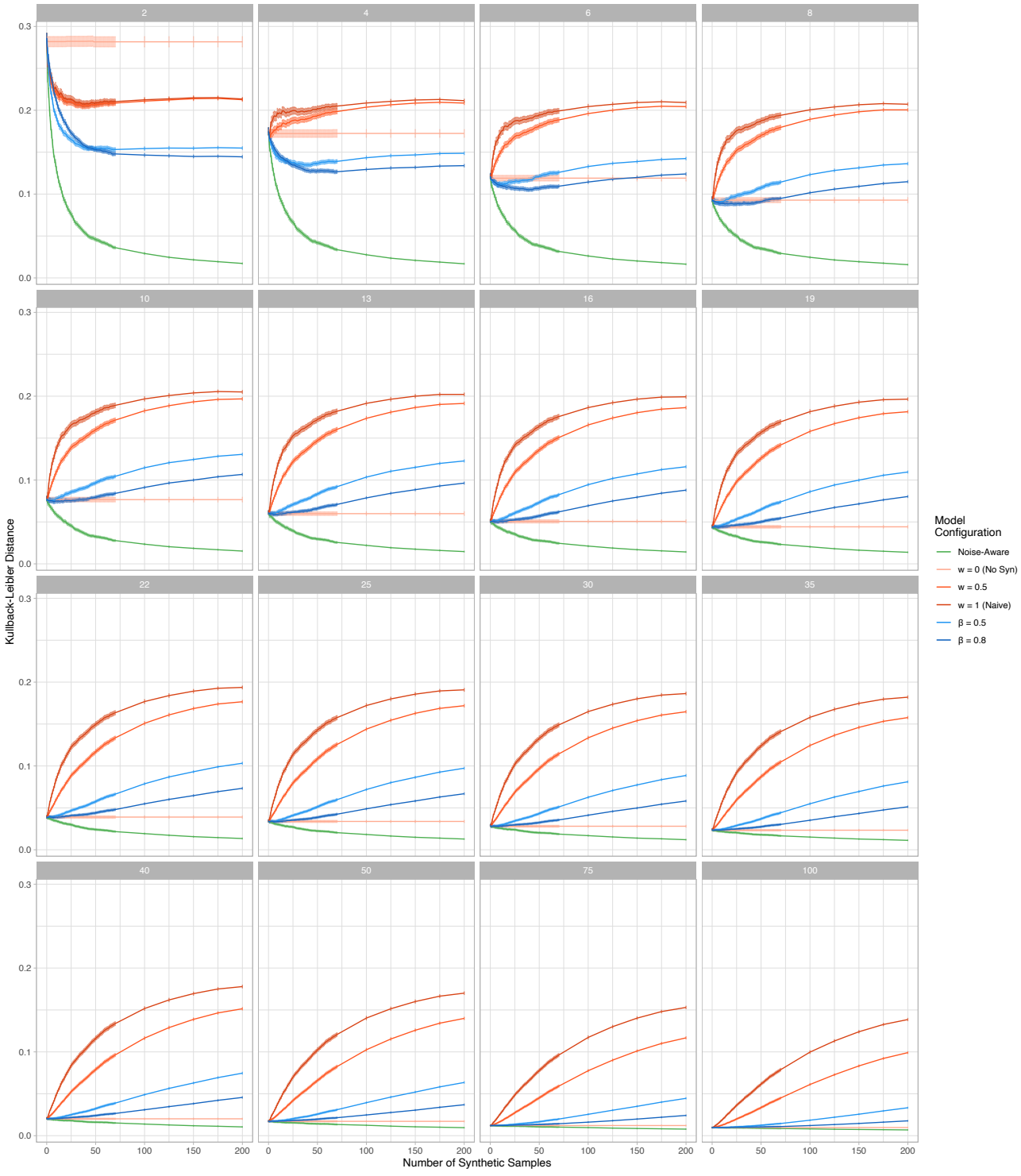

Figure A.18: Model comparison plots for each real data quantity n_L in the case of the simulated Gaussian experiments illustrating the KLD against the number of synthetic samples where DP of $\varepsilon = 6$ is achieved by the Laplace mechanism via noise of scale $\lambda = 1.0$.

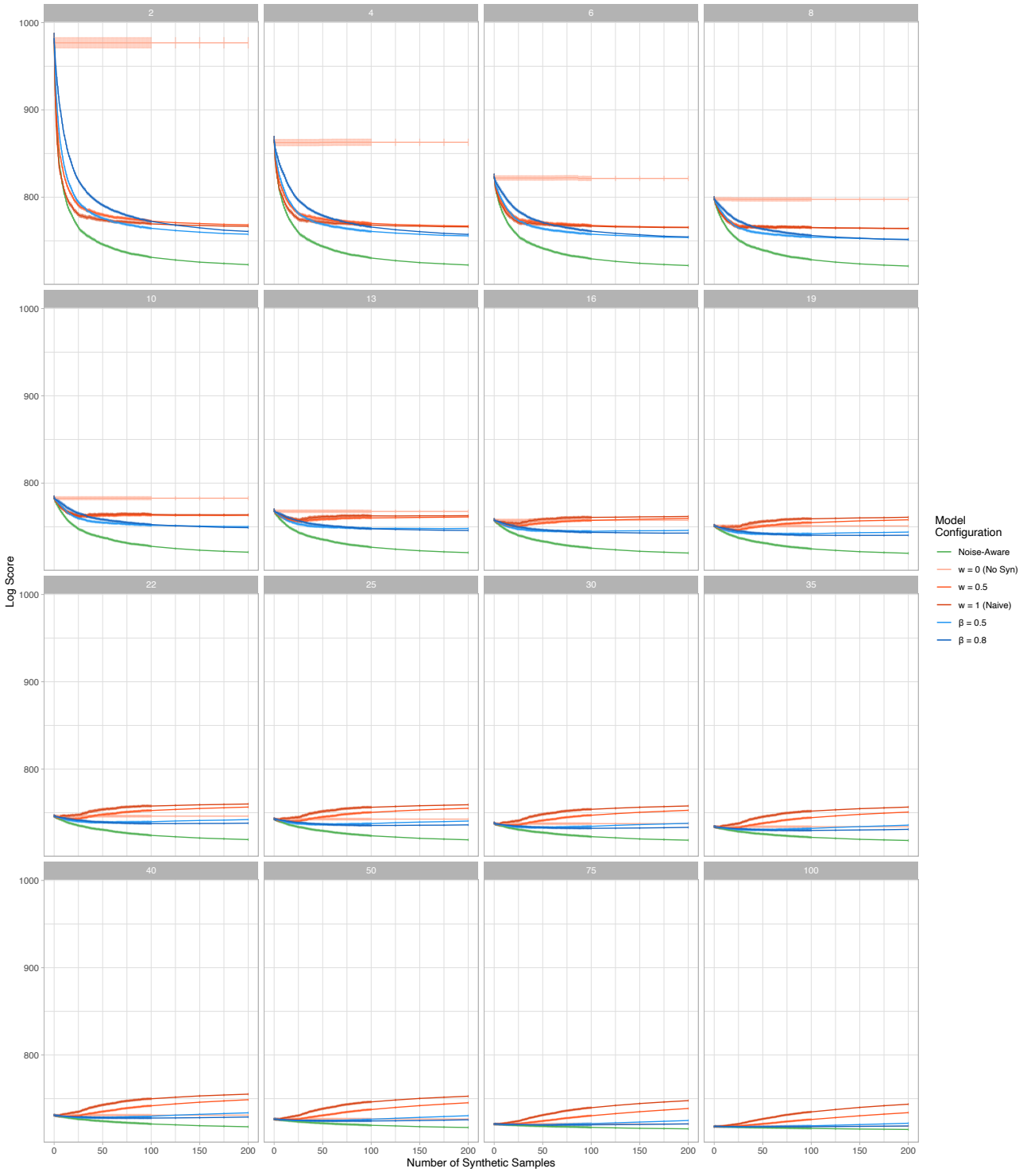

Figure A.19: Model comparison plots for each real data quantity n_L in the case of the simulated Gaussian experiments illustrating the log score against the number of synthetic samples where DP of $\varepsilon = 8$ is achieved by the Laplace mechanism via noise of scale $\lambda = 0.75$.

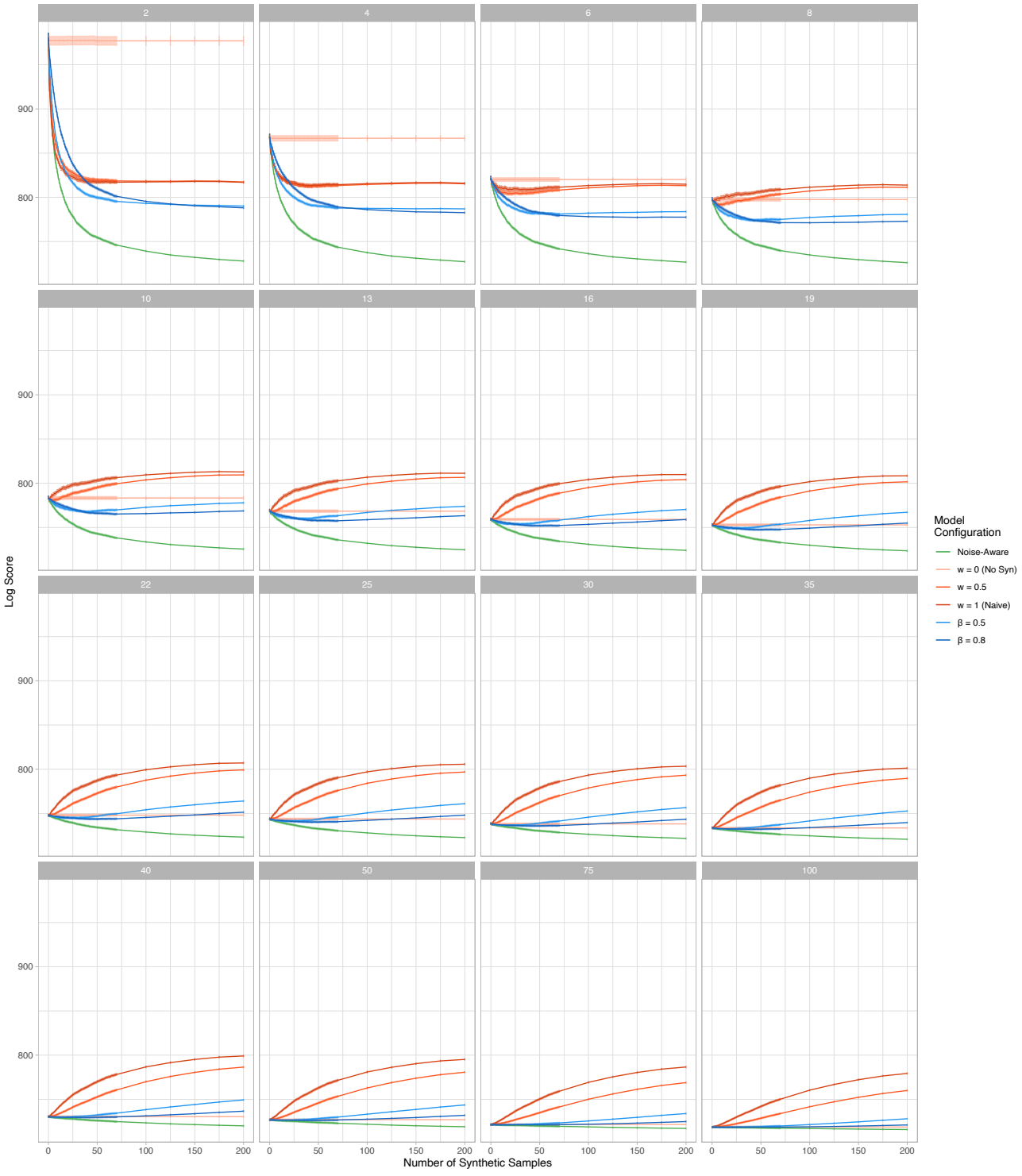

Figure A.20: Model comparison plots for each real data quantity n_L in the case of the simulated Gaussian experiments illustrating the log score against the number of synthetic samples where DP of $\varepsilon = 6$ is achieved by the Laplace mechanism via noise of scale $\lambda = 1.0$.

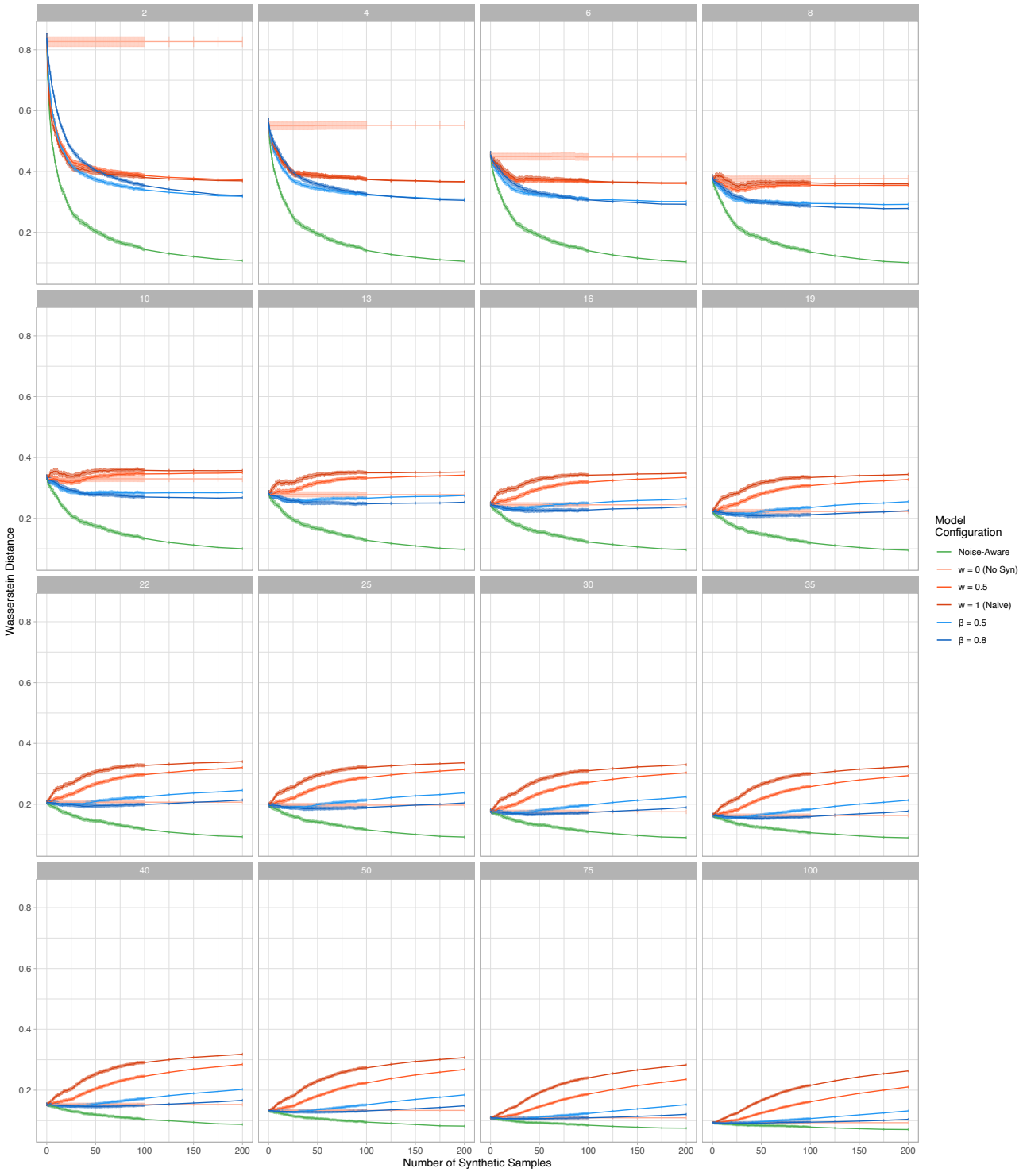

Figure A.21: Model comparison plots for each real data quantity n_L in the case of the simulated Gaussian experiments illustrating the Wasserstein distance against the number of synthetic samples where DP of $\varepsilon = 8$ is achieved by the Laplace mechanism via noise of scale $\lambda = 0.75$.

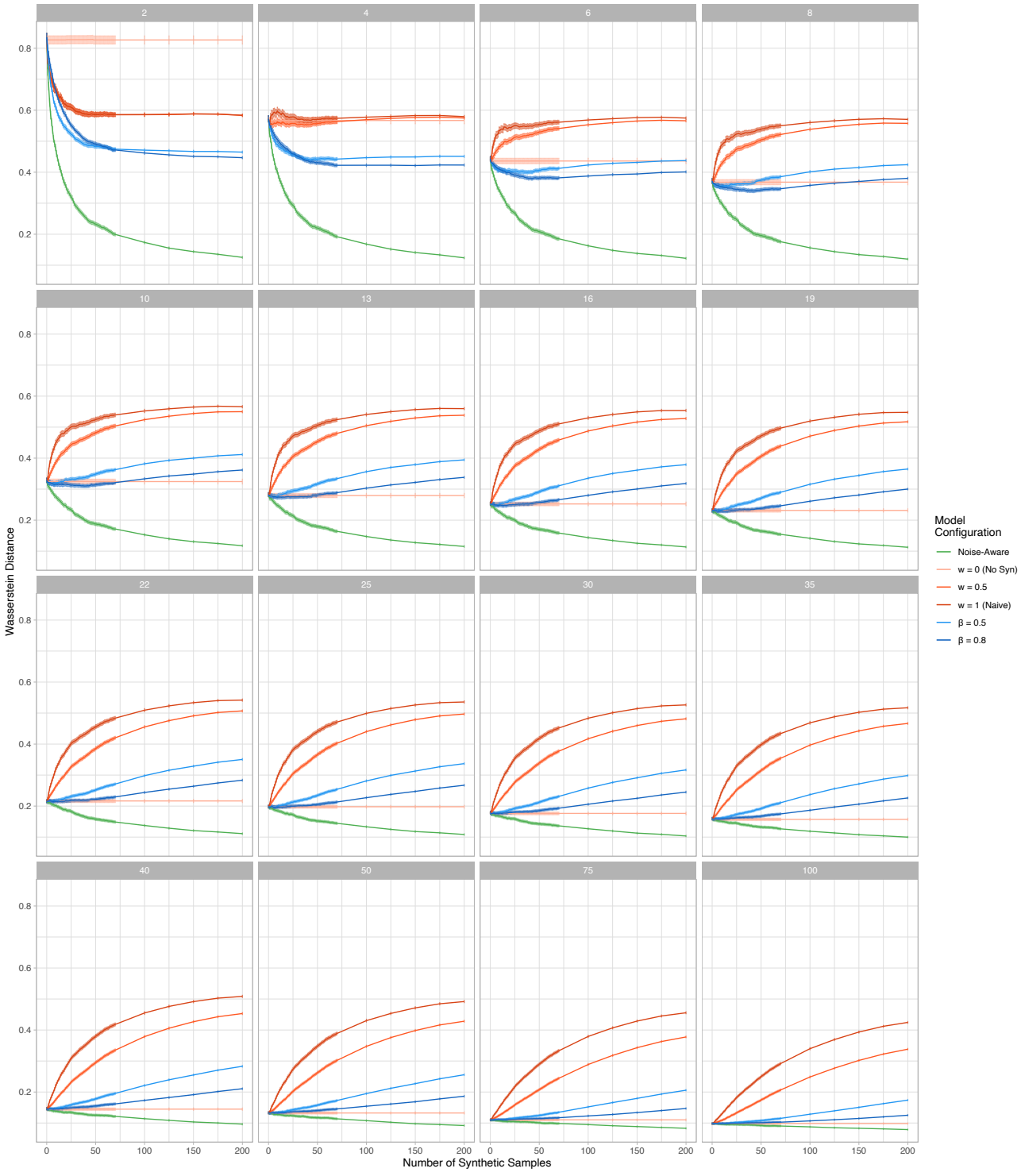

Figure A.22: Model comparison plots for each real data quantity n_L in the case of the simulated Gaussian experiments illustrating the Wasserstein distance against the number of synthetic samples where DP of $\varepsilon = 6$ is achieved by the Laplace mechanism via noise of scale $\lambda = 1.0$.

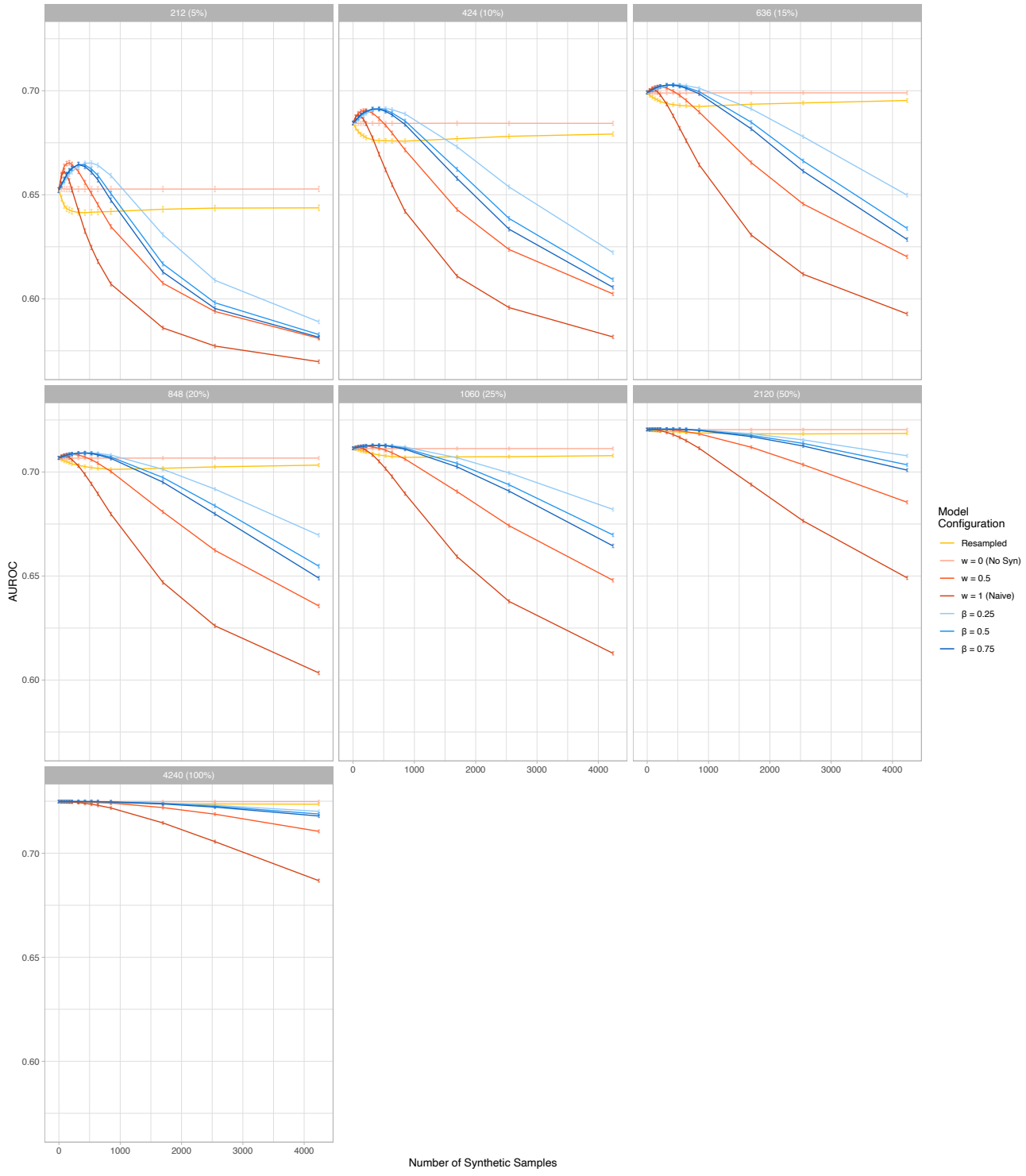

Figure A.23: Model comparison plots for each real data quantity n_L in the case of the logistic regression experiments on the Framingham dataset illustrating the AUROC against the number of synthetic samples where DP of $\varepsilon = 6$ is achieved via generation of synthetic datasets using the PATE-GAN.

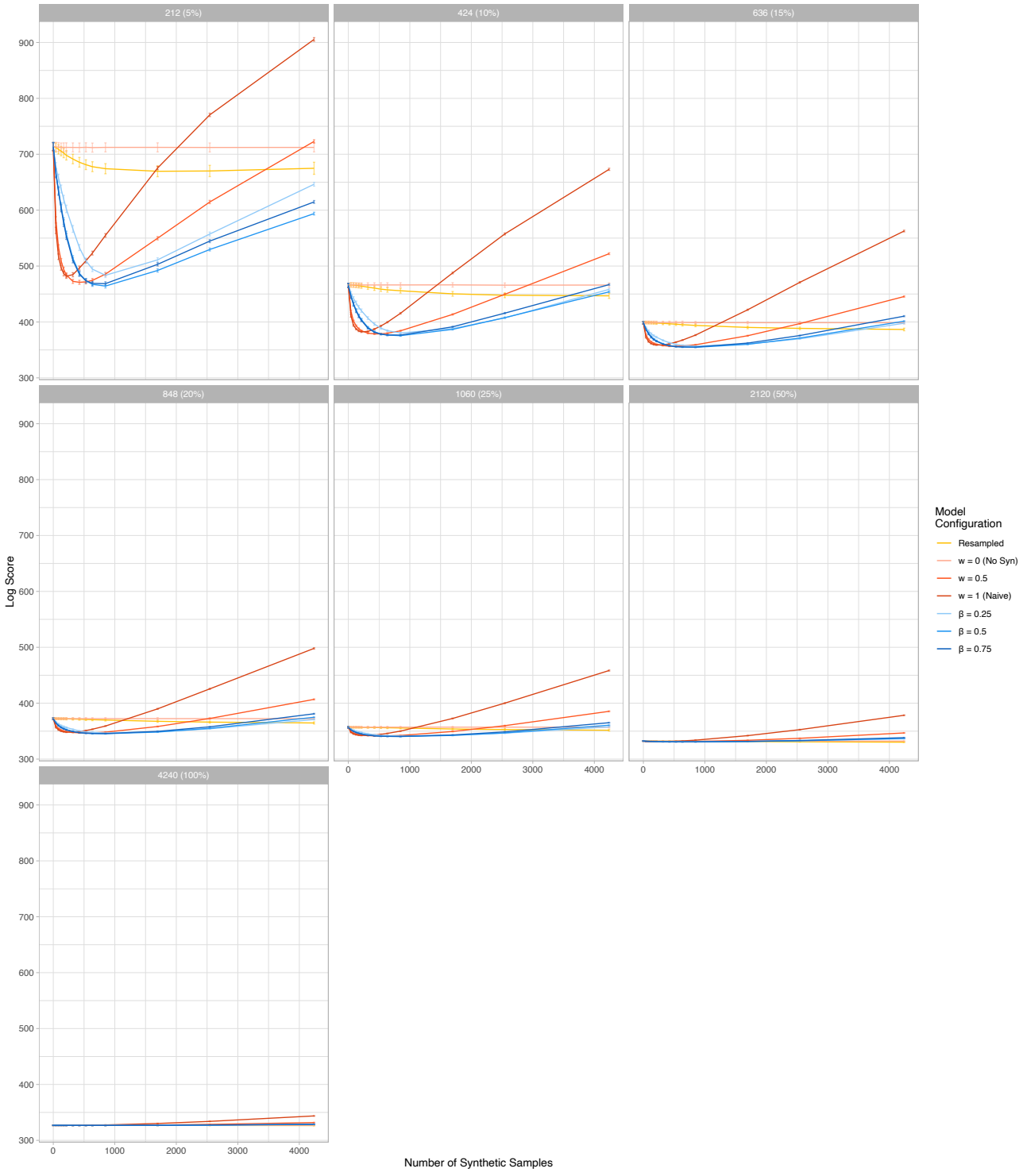

Figure A.24: Model comparison plots for each real data quantity n_L in the case of the logistic regression experiments on the Framingham dataset illustrating the log score against the number of synthetic samples where DP of $\varepsilon = 6$ is achieved via generation of synthetic datasets using the PATE-GAN.

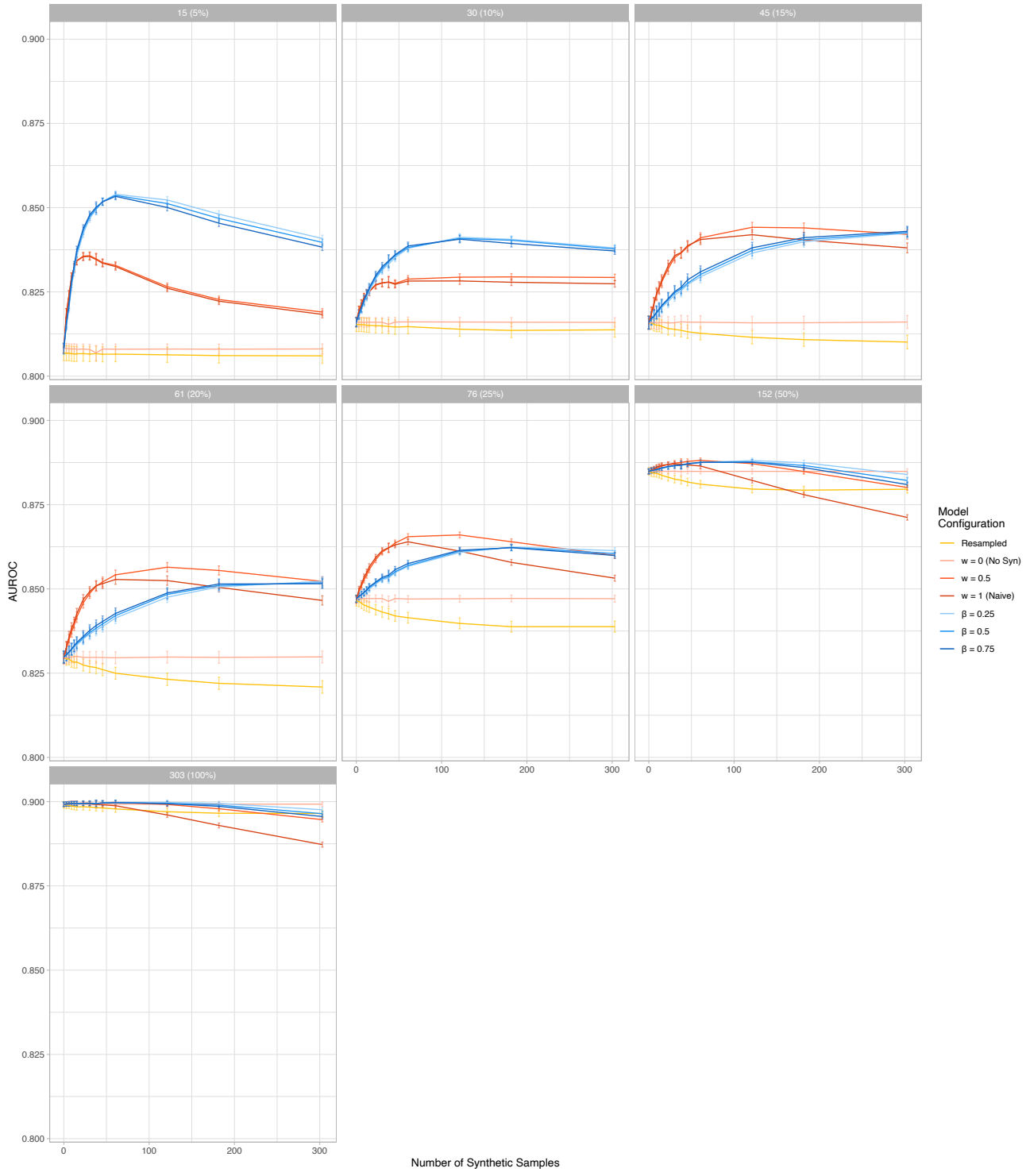

Figure A.25: Model comparison plots for each real data quantity n_L in the case of the logistic regression experiments on the UCI Heart dataset illustrating the AUROC against the number of synthetic samples where DP of $\varepsilon = 6$ is achieved via generation of synthetic datasets using the PATE-GAN.

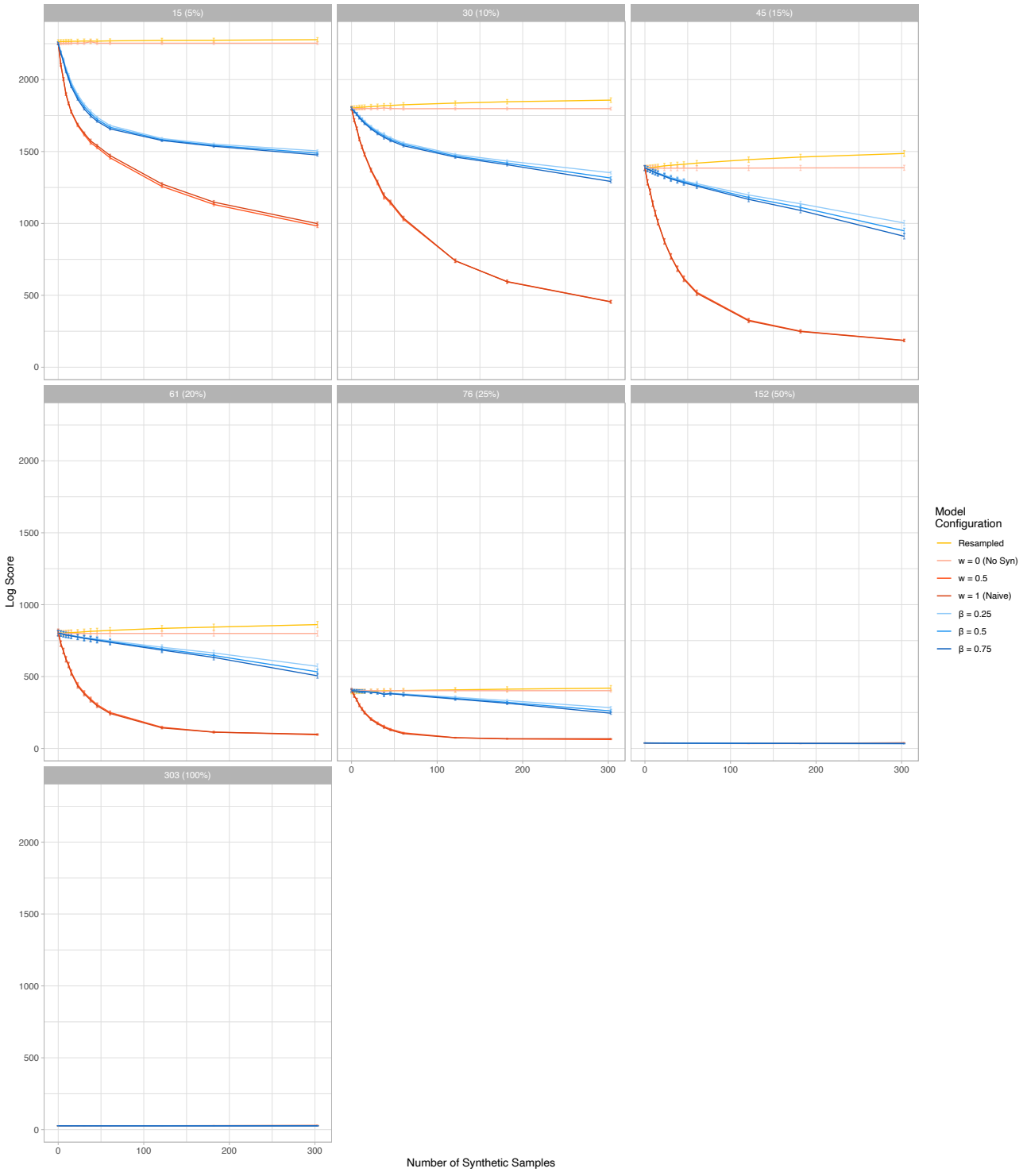

Figure A.26: Model comparison plots for each real data quantity n_L in the case of the logistic regression experiments on the UCI Heart dataset illustrating the log score against the number of synthetic samples where DP of $\varepsilon = 6$ is achieved via generation of synthetic datasets using the PATE-GAN.

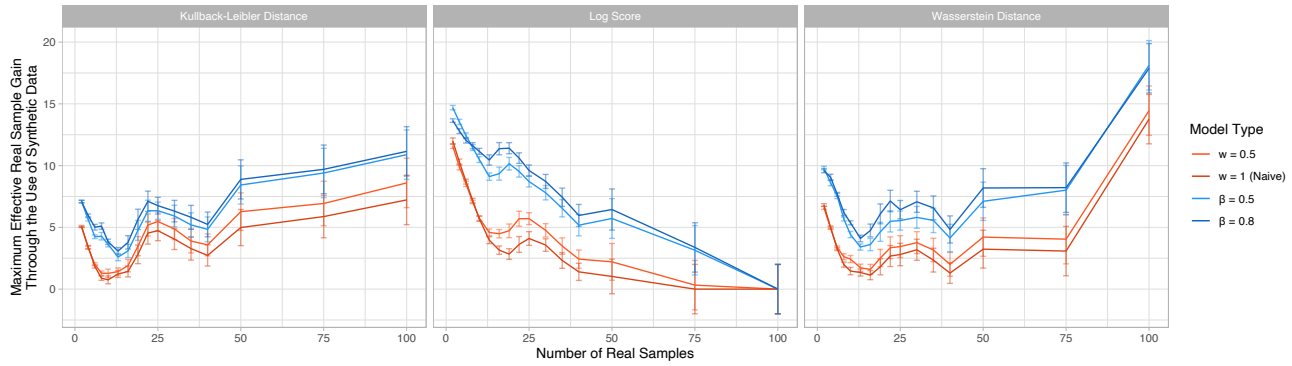

Figure A.27: n-effective plots plots for each of the relevant criteria in the case of the simulated Gaussian experiments illustrating the effective number of real samples to be gained through the use of synthetic data at each amount of real data n_L where DP of $\varepsilon = 8$ is achieved by the Laplace mechanism via noise of scale $\lambda = 0.75$.

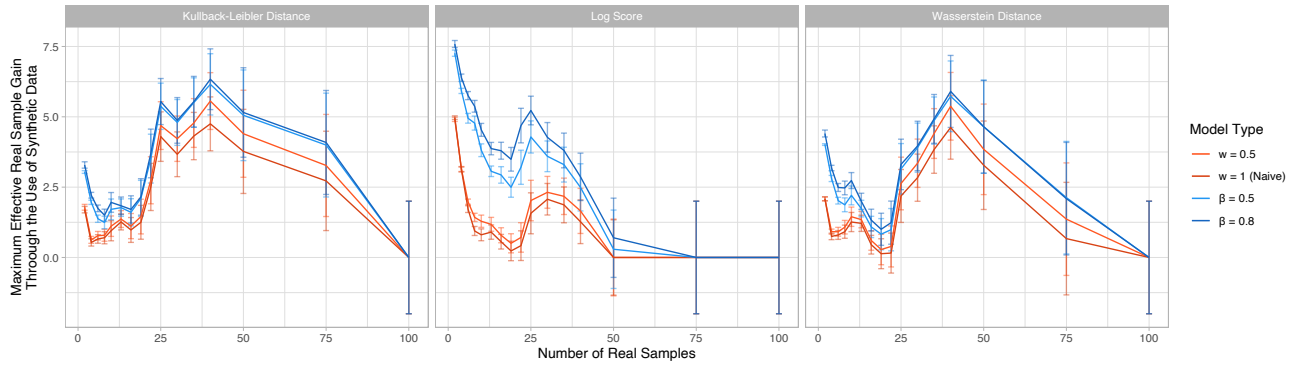

Figure A.28: n-effective plots plots for each of the relevant criteria in the case of the simulated Gaussian experiments illustrating the effective number of real samples to be gained through the use of synthetic data at each amount of real data n_L where DP of $\varepsilon = 6$ is achieved by the Laplace mechanism via noise of scale $\lambda = 1.0$.

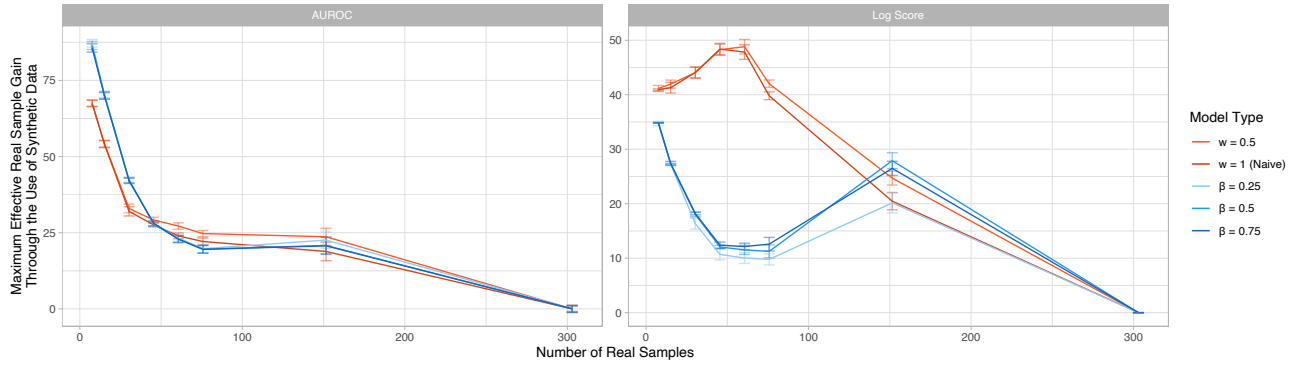

Figure A.29: n-effective plots plots for each of the relevant criteria in the case of the logistic regression experiments on the UCI Heart dataset illustrating the effective number of real samples to be gained through the use of synthetic data at each amount of real data n_L where DP of $\varepsilon = 6$ is achieved via generation of synthetic datasets using the PATE-GAN.

References

Amini, Z. and Rabbani, H. (2017). Letter to the editor: Correction to "the normal-laplace distribution and its relatives". *Communications in Statistics-Theory and Methods*, 46(4):2076–2078.

Basu, A., Harris, I. R., Hjort, N. L., and Jones, M. (1998). Robust and efficient estimation by minimising a density power divergence. *Biometrika*, 85(3):549–559.

Berk, R. H. et al. (1966). Limiting behavior of posterior distributions when the model is incorrect. *The Annals of Mathematical Statistics*, 37(1):51–58.

Bernstein, G. and Sheldon, D. R. (2018). Differentially private bayesian inference for exponential families. In *Advances in Neural Information Processing Systems*, pages 2919–2929.

Bissiri, P., Holmes, C., and Walker, S. G. (2016). A general framework for updating belief distributions. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*.

Blaom, A. D., Kiraly, F., Lienart, T., Simillides, Y., Arenas, D., and Vollmer, S. J. (2020). MLJ: A Julia package for composable Machine Learning.

Calders, T. and Jaroszewicz, S. (2007). Efficient AUC optimization for classification. In *Knowledge Discovery in Databases: PKDD 2007*, pages 42–53. Springer Berlin Heidelberg.

Carpenter, B., Gelman, A., Hoffman, M. D., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M., Guo, J., Li, P., and Riddell, A. (2017). Stan: A probabilistic programming language. *Journal of statistical software*, 76(1).

Dwork, C., McSherry, F., Nissim, K., and Smith, A. (2006). Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*, pages 265–284. Springer.

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

Efron, B. and Tibshirani, R. J. (1994). *An introduction to the bootstrap*. CRC press.

Ge, H., Xu, K., and Ghahramani, Z. (2018). Turing: a language for flexible probabilistic inference. In *International Conference on Artificial Intelligence and Statistics, AISTATS 2018, 9-11 April 2018, Playa Blanca, Lanzarote, Canary Islands, Spain*, pages 1682–1690.

Ghosh, A. and Basu, A. (2016). Robust bayes estimation using the density power divergence. *Annals of the Institute of Statistical Mathematics*, 68(2):413–437.

Gneiting, T. and Raftery, A. E. (2007). Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378.

Hoffman, M. D. and Gelman, A. (2014). The no-u-turn sampler: adaptively setting path lengths in hamiltonian monte carlo. *J. Mach. Learn. Res.*, 15(1):1593–1623.

Jewson, J., Smith, J., and Holmes, C. (2018). Principles of Bayesian inference using general divergence criteria. *Entropy*, 20(6):442.

Jordon, J., Yoon, J., and van der Schaar, M. (2018). Pate-gan: Generating synthetic data with differential privacy guarantees. In *International Conference on Learning Representations*.

Kurtek, S. and Bharath, K. (2015). Bayesian sensitivity analysis with the fisher–rao metric. *Biometrika*, 102(3):601–616.

Meinshausen, N., Meier, L., and Bühlmann, P. (2009). P-values for high-dimensional regression. *Journal of the American Statistical Association*, 104(488):1671–1681.

Reed, W. J. (2006). The normal-laplace distribution and its relatives. In *Advances in distribution theory, order statistics, and inference*, pages 61–74. Springer.

Rueshendorff, L. (1977). Wasserstein metric. `http://encyclopediaofmath.org/index.php?title=Wasserstein_metric&oldid=50083`. Accessed: 2020-10-22.

Watson, J. A. and Holmes, C. C. (2020). Machine learning analysis plans for randomised controlled trials: detecting treatment effect heterogeneity with strict control of type i error. *Trials*, 21(1):156.

Xie, L., Lin, K., Wang, S., Wang, F., and Zhou, J. (2018). Differentially private generative adversarial network. *arXiv preprint arXiv:1802.06739*.