We thank Reviewers (R) 1, 2, 3, 4 and 5 (who gave us marks 7, 6, 8, 6, and 6, respectively) for their pertinent remarks.

R2+R4 (also R1+R5): Main contribution. We recall our main contribution. Recall that prox_G generalizes the projection onto convex set in a way that $\operatorname{prox}_G(x) \in \operatorname{dom}(G)$. Importantly, in PSGLA *the iterates are feasible*: $x^{k+1} \in \operatorname{dom}(G) = \operatorname{supp}(\mu^\star)$, contrary to alternative methods. The PSGLA method proposed in this paper *extends in a natural way* particular cases that were considered in the literature, namely i) the case where G is Lipschitz [18, Sec 4.2] (in particular $\operatorname{supp}(\mu^\star) = \mathsf{X}$) and ii) the case where G is the indicator of a convex compact set [9] (in particular $\operatorname{supp}(\mu^\star) \neq \mathsf{X}$), in which case PSGLA has a high complexity $\mathcal{O}(1/\varepsilon^{12})$. All the other cases, where G is a general convex l.s.c, potentially with a domain $\operatorname{dom}(G) \neq \mathsf{X}$ (for instance the G considered in the experiments l.456) where not analyzed before. *Our main contribution is to analyze PSGLA in these new cases.* This is a challenging problem: we had to develop new mathematical tools (e.g. the duality gap arising from the primal dual interpretation of PSGLA) for the analysis. Using these tools, we obtained surprising results: although G can have a domain ($\operatorname{supp}(\mu^\star) \neq \mathsf{X}$), the complexity of PSGLA in these new cases is basically the same as in the case where G is Lipschitz ($\mathcal{O}(1/\varepsilon^2)$).

R1+R3+R4: Numerical experiments. Although our paper is mainly of a theoretical nature, we provided detailed numerical experiments + the associated code. We will use the 9th page to move some experiments to the main text if our paper is accepted. We do not believe that our experiments are too simple as R4 says. We considered a sampling problem in a multidimensional half space of matrices relevant in the field of random matrices. The function G is given in l.456 and the computation of prox_G in closed form relies on recent results on proximity operators. Particular cases of PSGLA considered previously are not able to tackle this sampling problem. The main message of the experiment section is: PSGLA only produces iterates in the support of $\mu^\star$, contrary to alternative methods.

R1: 1) G is a general convex l.s.c. but using [24, Th 25.5], G is almost surely differentiable on its domain, therefore the integral is well defined. 2) SGD can be written $x^{k+1} = x^k - \gamma \nabla F(x^k) + \gamma w^{k+1}$, where w^{k+1} is a martingale increment. In Langevin the noise W^k is scaled by $\sqrt{\gamma}$ instead. There is more noise in Langevin, that's why Langevin explores the whole support of $\mu^\star$, whereas SGD converges to $\arg\max \mu^\star$. 3) Yes, and we did it! In Appendix D, we used (cheap) stochastic proximity operators instead of full proximity operators and showed that the convergence rates remain unchanged. Another approach (that we will acknowledge) could be to adapt to our setting the proof technique of Pesquet and Combettes in a series of papers on primal dual optimization with approximate proximity operators. Note however that many proximity operators can be computed in closed form (hinge loss, logistic loss, many penalizations, etc., see also the experiments l.456) thanks to research efforts on this topic, see proximity-operator.net. 4) Convexity is needed to prove that the duality gap is nonnegative (Th3) which is fundamental in our approach (We iterated (27)). For nonconvex cases, see [30,33] that made nice connections between nonconvex Langevin and nonconvex optimization.

R2: 1) We think that R2 has missed a key part of our main contribution, see above. 2) In the literature on Langevin algorithm, complexity results are indeed often expressed in terms of the parameters of the problem like the Lipschitz constants and d. If G is M-Lipschitz, [18, discussion after Corollary 18] gives a complexity result for PSGLA in terms of σ_F^2, L, M, d by using an approach similar to [[1, Lemma 5]]. Although the constants σ_F^2, L, M implicitly depend on d, they have an explicit meaning. Using [[1, Lemma 5]] in our case, we can obtain the same complexity result as in [18], but by replacing M^2 by $I := \int \|\nabla G\|^2 d\mu^\star$. We will acknowledge that I is a bit less intuitive than M, but $\sqrt{I}$ can be seen as a generalized Lipschitz constant since $I \leq M^2$ if G is M-Lipschitz. 3) $x^\star$ is a measurable map $\Omega \to \mathsf{X}$, i.e., a random vector (see l.246). Since $x^\star : (x, y) \mapsto x$ and $\Omega = \mathsf{X}^2$ is endowed with $\pi^\star$, the distribution of $x^\star$ is the first marginal of $\pi^\star$, i.e., $\mu^\star$ (such construction is sometimes used in probability theory). 4) l.303: we will acknowledge [[1]] (we did not know). 5) The meaning of the convergence of the duality gap to zero is an important interesting question that is the subject to further research. We are specifically interested in understanding its relationship with weak or KL convergence. It is also a difficult question, and we don't have a clear answer for projected Langevin. In this paper, we focused on the convergence in W_2 in the case where F is strongly convex and G general convex l.s.c, in order to cover new cases.

R1+R3: Thank you for your positive feedback. We will use the 9th page to move some experiments to the main paper.

R1+R4+R5 (also R3): Intuition. Regarding the FB, everything is made rigorous in Lemma 4. We provided a whole Appendix (C) to provide intuition on the relationship on primal dual convex optimization, FB and sampling, and how we had the idea of using a primal dual view to obtain our complexity results. Notably, our Lagrangian is similar to the one used in optimization, see C.2. We will go further by moving some material from C.2 to the introduction.

R4: l.21, l.170: we corrected. l.198: (12) is valid only for any μ', but only using $T_\mu^{\mu'}$ the *optimal* pushforward. This is the main difference with convexity, that makes things more challenging. Th2: w is a (random) vector because the elements of the subdifferential are (random) vectors (as in optimization), see l.195. Th2: If $V \equiv 0$, $w\rho = \nabla\rho$ i.e. $w = \nabla \log(\rho)$. Then we take $\rho = \exp(-V)$ which gives the result. γ: There is a trade-off: if γ is small, MYULA (or any Langevin algorithm) is more precise but slower to reach its invariant distribution. That's why we compared MYULA vs PSGLA at equal learning rate γ.