Thanks for the helpful feedback. We got a clear sense of where more clarification would be helpful.

General Discussion Our work is part of the following larger and important discussion within the NeurIPS community: To what solution do neural nets (trained w. GD) converge to in the overparameterized setting? This discussion was reignited by two recent NeurIPS papers (Gunasekar et al. 2017) and (Vaskevicius et al. 2019). They showed that in a certain context, continuous GD updates converge to sparse solutions. In a recent COLT paper (Amid, Warmuth 2020) this was related to a particular MD algorithm: if the edges of a linear neuron are doubled (as in Fig. 2), then continuous GD on this network simulates the unnormalized exponentiated gradient algorithm (EGU). EGU is the classical MD algorithm (with the log link) and there is a long history of results showing that it does well when the solution is sparse. Previously it was thought that GD cannot take advantage of the sparsity of the solution.

In this paper we show that the reach of GD is even much farther: Any CMD update can be simulated by another CMD provided a certain reparameterization function exists. We give many reparameterization examples in the paper and in particular, we develop a family of MD updates (using the $\log_\tau$ link) that lie between GD and EGU using vanilla GD (by means of a reparameterization).

What is the surprising insight? The structure of the network is key! Much research has been done in exploring multi layer nets with fully connected (FC) layers. In particular, wide FC layers have been investigated. For example the NeurIPS paper, Arora et al. "On Exact Computation with an Infinitely Wide Neural Net." NeurIPS 19, shows that an infinitely wide neural net behaves like a kernel method.

Our research leads us in a different direction in that a linear neuron in which the edges are doubled (Fig. 2) can exploit sparsity. In particular, combined with some well known bounds for any kernel method, this shows that GD on the thin network of Fig. 2. can learn certain sparse problems exponentially faster than any kernel method. We believe that this discussion is central to neural net research and important to the NeurIPS community.

We will add more context along the above to the final version. We will also smoothen the transitions and give more extensive definitions as suggested by the reviewers. The discussion of duality is important for the optimization context. We will relegate much of that to the appendix in exchange for giving more motivation for the main results of the paper.

Reviewer #2 • *"given that this work is being submitted to NeurIPS, I don't see enough of a relationship to artificial neural networks... Describing the specifics of using this work to help the neural processing world feels as though it would be more appropriate."*

Please note that many similar theoretical work such as (Gunasekar et al. 2017) and (Vaskevicius et al. 2019) have appeared before in NeurIPS. That is, NeurIPS has always been one of the top venues for publishing results on learning theory and optimization. The current submission is a theoretical work, but also has practical implications for understanding the training dynamics of deep neural networks. Therefore, we would like to encourage Reviewer #2 to cast their judgment solely based on the quality of the paper and significance of the theoretical results and defer the concerns about the relevance of the work to the NeurIPS community to the ACs.

Reviewer #4 • *Lack of definitions:* We will add the missing definitions and improve the flow.

• *Meaning of 'independent variables:* We refer to independence by means of elementary calculus. For independent variables, a variable in an equation may have its value freely chosen without considering values of the other variables.

• *Motivation of the dual form:* The dual form of MD is extremely important for efficient implementation of updates and analyzing the worst case regret bounds. We will make the motivation clear. We will also improve the write up to ease the transition between the sections.

• *"Why should we care about reparameterized CMD as GD?"*

For a long time, many MD updates such as EGU had been considered fundamentally different than GD (multiplicative vs. additive). Therefore, the EGU update was considered to be unrealizable by additive updates such as backprop. We show that EGU (and many other CMD) updates are realizable "exactly" via backprop on a reparameterized network. Following (Amid, Warmuth 2020), we believe that the discretized updates will closely mimic the original updates. This has important implications for implementing these updates via GD in the existing platforms (such as TF).

Reviewer #5 • *"the authors show that CMD can be presented as minimizing a trade-off between the loss and a Bregman momentum. ... I am unsure about the significance of this contribution."*

Prior to this work, CMD was motivated as the limit point of MD when the step-size goes to zero. Alternatively, we motivate CMD as the minimization of a functional objective. This novel view of CMD is useful for deriving worst case regret bounds in the continuous-time, which we defer to future work. This also allows a novel approach for deriving MD updates by directly discretizing the functional objective (please see Appendix C for more details).