Overall: We thank the reviewers for their comments and questions. We are encouraged to see that all reviewers recommended to accept the paper. The one repeated criticism is the simplicity of the model we study, and in particular the use of a firing-rate based dynamics rather than spiking neurons. It is an understandable question, in particular when the framework we present is conceptually similar to previous studies on efficient coding of spiking networks [Boerlin *et al.*, 2013]. When considering the work on spiking networks, it is tempting to think that the superclassical efficiency, namely the scaling of the readout error as $1/N$, is a consequence of the spiking dynamics. Instead we consider a more *general* model and assert that the efficiency is the result of the strong negative feedback that scales as $b \sim N$. Furthermore, our theory is agnostic towards the local nonlinear transfer function. Thus, we believe that the simplicity of the model is rather a strength, as it strips the framework only to the necessary components. It is a fair question to ask if our results hold for spiking dynamics as well. As a first approximation, one can consider a spiking neuron as a renewal process with Poisson statistics where the Poisson mean is the instantaneous firing rate and is given by $\phi(x_i(t))$. The transfer function here could be the mean-field firing-rate approximation for the spiking model. Here, it is easy to be convinced that the spiking dynamics is equivalent to an extra noise term [Kadmon & Sompolinsky, 2015]. As mentioned in the paper, numerical simulations of the efficient coding framework with Leaky Integrate and Fire (LIF) neuron show similar qualitative results [Chalk *et al.*, 2016]; these observation were not understood theoretically. We note that we have derived a rigorous theory for LIF Neurons. However, the spiking dynamics require a different mean-filed approach which is out of the scope of the current paper. We will publish this theory elsewhere as it does not undermine the novelty nor the significance of the more general results we present here.

In the following we address other concerns and questions raised by the reviewers.

Reviewer 1: In addition to the concern on the rate-based dynamics, which we have addressed above, the reviewer noted that we did not include numerical simulation for time-dependent and high-dimensional signals. The treatment of these was delegated to the supplementary material (SM), together with the encoding of autonomous dynamical systems as they do not provide additional insights. However, we agree with the reviewer that we can add a figure demonstrating the theory for high dimensional dynamic signals to the SM. On the camera ready version of the paper we have slightly more space, and we can increase the size of figures and fonts to make them more legible, following the reviewers comments. The references on predictive coding will be also corrected.

Reviewer 2: The main goal of our theory is to understand how well a large network of unreliable units can faithfully represent a signal. We did not attempt to claim that the simple setting we use preforms complex computations. It is actually a well known criticism of Balance Network, that they can represent only linear transformation [Ahmadian & Miller, 2019]. However, it does not mean that the network cannot perform more complex computations. In the SM we show how the network can implement a general autonomous linear dynamical system. In [Alemi *et al.*, 2017] a similar framework was used to encode nonlinear autonomous dynamics. Finally, a recent work has shown that Balanced Networks can implement rectified-linear transformations and be used to perform a variety of nonlinear computations [Baker, Zhu & Rosenbaum, 2020]. Nevertheless, the question of how the computation can be reliably encoded by the population is ubiquitous, and that is the problem we address in this work. Following the reviewers comment, we will emphasize in the discussion the relevance of our work for nonlinear computations.

Reviewer 3: We thank the reviewer for noting that we did not emphasize enough the difference in scaling of the readout error. In particular, a central point is that in the case of tight balance [Deneve & Machens, 2016], the negative feedback scales as $b \sim N$. We emphasize that previous studies achieve superclassical scaling of error ($1/N$) because of the scaling of feedback. We believe that showing that the superclassical error can be obtained in rate-based network with arbitrary nonlinear neurons is rather a strength of our theory.

The reviewer mentioned that previous works have decoupled the random connectivity from the ordered part (e.g. [Mastrogiuseppe & Ostojic, 2018]). However, they did not use them in a strong balance setting as a mean to decouple the random fluctuations from the magnitude of structured part. We will elucidate our approach in the text. Lastly, the reviewer asks about the difficulties of the full chaotic solution. Indeed, the autocorrelation of the chaotic fluctuation has been calculated before by several authors. However, to solve for the fluctuations of the *readout*, the autocorrelation of the chaotic fluctuations is used as a noise source in the ODE for the decoder in Eq. (9). An exact solution for the chaotic autocorrelation is usually found numerically by solving the mean-field equations. Thus, the exact expression for the readout error can only be evaluated numerically. In order to gain insight into the solution, we propose a simple approximation that captures the correct error scaling, and reveals the mechanism by which the negative feedback efficiently suppress chaotic fluctuations, namely the low-pass filter on the fluctuations.

Reviewer 4: The reviewer correctly notes feedforward input weights and recurrent connectivity are closely tied to ensure the balanced cancellation of feedforward input by the recurrent connectivity as in many predictive coding frameworks like [Deneve & Machens, 2016]. Hebbian learning of recurrent synapses driven by feedforward weights $\mathbf{w}$ could learn the recurrent part $\mathbf{w}\mathbf{w}^{\mathbf{T}}$. Any errors in the learned weights are accounted for by the random part $\mathcal{J}$.