First steps to maybe understand types of autoencoder strategies as phase transitions in a chanaging distribution

Samuel Jack Jolley, under the supervision of Daniel Murfet, in Janurary and Feburary 2023 ~ contact: samuel.jolley@anu.edu.au

Introduction in Singular Learning Theory [2]

Singular learning theory considers models which are **singular** (defined shortly), and especially SLT corrects misapplications of theory which is valid for non-singular models but erronous in the singular case. Additionally, SLT provides alternative and extended theory for singular models.

A model attemping to fit a true distribution $q(y|x)q(x) = q(y,x)$ paramatrised like $\{p(y|x,\theta) \;:\; \theta \in \Theta\}$

is **identifiable** if the mapping

$$\theta \to p(y \mid x, \theta)$$

is injective.

We should also consider the FIsher Information matirx, a function on the paramaters of a model theta,

$$I(\theta)_{ij} = \int\int \frac{\delta}{\delta\theta_i}[log(p(y|x,\theta))]\frac{\delta}{\delta\theta_j}[log(p(y|x,\theta))]q(y|x)q(x)\,dxdy$$

Then, a model is **regular** if it is both identifiable and has positive definite Fisher information matirx. It is **strictly singular** if it not regular.

Objects of study is SLT are conviently phrased as tuples

$$(p(y,x,\theta), q(y,x), \varphi(\theta))$$

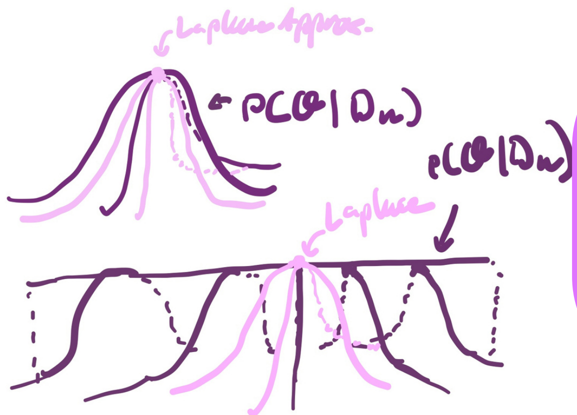
with $\varphi(\theta)$ a prior. The Kullback-Liebler divergence is a fundemental measure of a model's fit to the truth, which we use as a function of the model paramaters like

$$KL(\theta) = \int\int q(y|x) log\frac{q(y|x)}{p(y|x,\theta)}q(x)\,dxdy$$

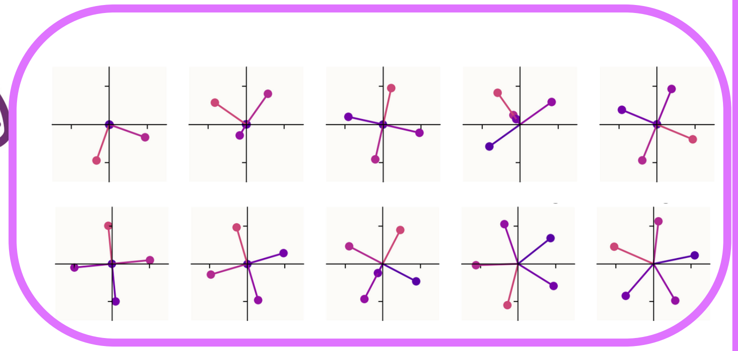
We can then write the posterior distribution on the paramater space for some stochastic dataset D_n drawn of the true distribution, and the corresponding empirical KL divergence like

$$p(\theta|D_n) = \frac{1}{Z}\varphi(\theta)\exp\{-nK_n(\theta)\}$$

Notably, non-trivial neural network are singular.



An example where SLT is needed: for singular models, fitting a Guassian to the MAP is not correct even in the limit. Inspired by [5].



Results of training the ReLU toy autoencoder with varying sparsity: lowest sprsity at top-left, decreases left to right, then decreases left to right along the bottom.

Toy model of an autoencoder [1]

In pursuit of a mechanistic understanding of neural networks, the paper [1] defines a toy model of an what is essentially an autoencoder, a artificial dataset with paramaters, and trains their model with SGD. This model has a feature space $\mathbb{R}^5$ and latent space $\mathbb{R}^2$.

The model is paramatrised by network weights W and biases, and these implement it as

$$f_\theta(x) = ReLU(W^TWx + b)$$

The dataset distribution is uniform over $[0,1]^5$, but with additional structure of a "sparsity" μ, which controls the likelihood some component of $\mathbb{R}^5$ will be zeroed.

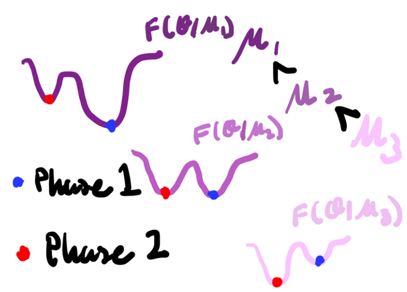
Loss is MSE weighted by an "importance" factor, which we set as $I_i = 0.9^i$, and training is done with Adam.

Attempting a synthesis

Interest spark: types of encoding

Something that seemed interesting about the toy model results was that for a gradually changing sparsity, the network seemed to learn distinct kinds of encoding: orthogonal, triplet, double othogonal and pentagonal.

Phase transiitions



The treatment in SLT of phase transitions [3,4] inspires the hypothesis to be tested here. I guess that as the sparsity paramter varies, different minima of the landscape associated with the KL divergence exchange role as global minima.

Adjustments to reconcile

The toy autoencoder is not phrased in the language of SLT, and this would be good ground work to do: we can construct a comparable SLT instance as

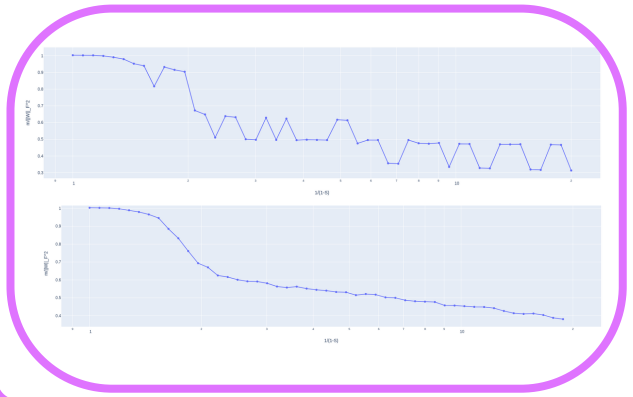
$$(p(y|x,\theta) = \frac{1}{(2\pi)^{\frac{n}{2}}}\exp\{-\frac{1}{2}||y - f_\theta(x)||^2\},\; q(y|x) = \frac{1}{(2\pi)^{\frac{n}{2}}}\exp\{-\frac{1}{2}||y - x||^2\},\; \varphi(\theta))$$

which has KL divergence equivilant (almost: we are ignoring "importance") to the loss:

$$K(\theta) \propto \int_X ||f_\theta(x) - x||^2 q(x)dx,\; K_n(\theta) = \frac{1}{n}\sum_{i=1}^{n}||f_\theta(x_i) - x_i||^2$$

Further, within the model, exhanging ReLU for paramatrised swish functions makes the model analytic.

Experiment and Results



Training the swish models as before, with a gamma value of 30 (high enough that trials are very similar to ReLU), with a granularity in sparsity from 0 to 1 of 50. Top is single, bottom is average of 100 runs. The vertical measure is $\frac{m}{||W||_F^2}$ for W the weights, which is intented to measure the "dimensions per feature".

Discussion and comments

Firstly it is worth explicitly saying that the SLT formalism is not really used. It offers conceptualisation, and if taken further might give a satisfying analytic / numerical approx result to tie in with the current experiment.

On the graphs: I would argue that the first, (the singular model) one shows that the paramaters have discrete types. The averaged graph looses this: I think this could be improved by taking only models with very low loss in the average

Acknowledgements

I would like to thank Daniel Murfet, for their insightful guidance and notes, their enthusiasm and interest, and their tolerance for my poor time management.
I would like to thank the team at UniMelb who put together this program: Roy Ridgway, Thomas Quella, and everyone.
Also thanks to [1] for publishing their code, which I used and adapted.

References

[1] ***Toy Models of Superposition,*** various authors from Anthropic and Harvard, 2022, available at https://transformer-circuits.pub/2022/toy_model/index.html

[2] ***Deep Learning is Singular, and That's Good***, Daniel Murfet, Susan Wei, Mingming Gong, Hui Li, Jesse Gell-Redman, Thomas Quella, 2020, available at https://arxiv.org/abs/2010.11560

[3] ***Phase Transitions in Neural Networks***, Liam Carroll, 2021, available at http://therisingsea.org/notes/MSc-Carroll.pdf

[4] ***Deep Learning Theory 3: Phase Transitions***, Daniel Murfet, 2020, available at http://www.therisingsea.org/notes/metauni/dlt3.pdf

[5] ***Neural networks generalize because of this one weird trick***, Jesse Hoogland, 2023, available at https://www.lesswrong.com/posts/fovfuFdpuEwQzJu2w/neural-networks-generalize-because-of-this-one-weird-trick