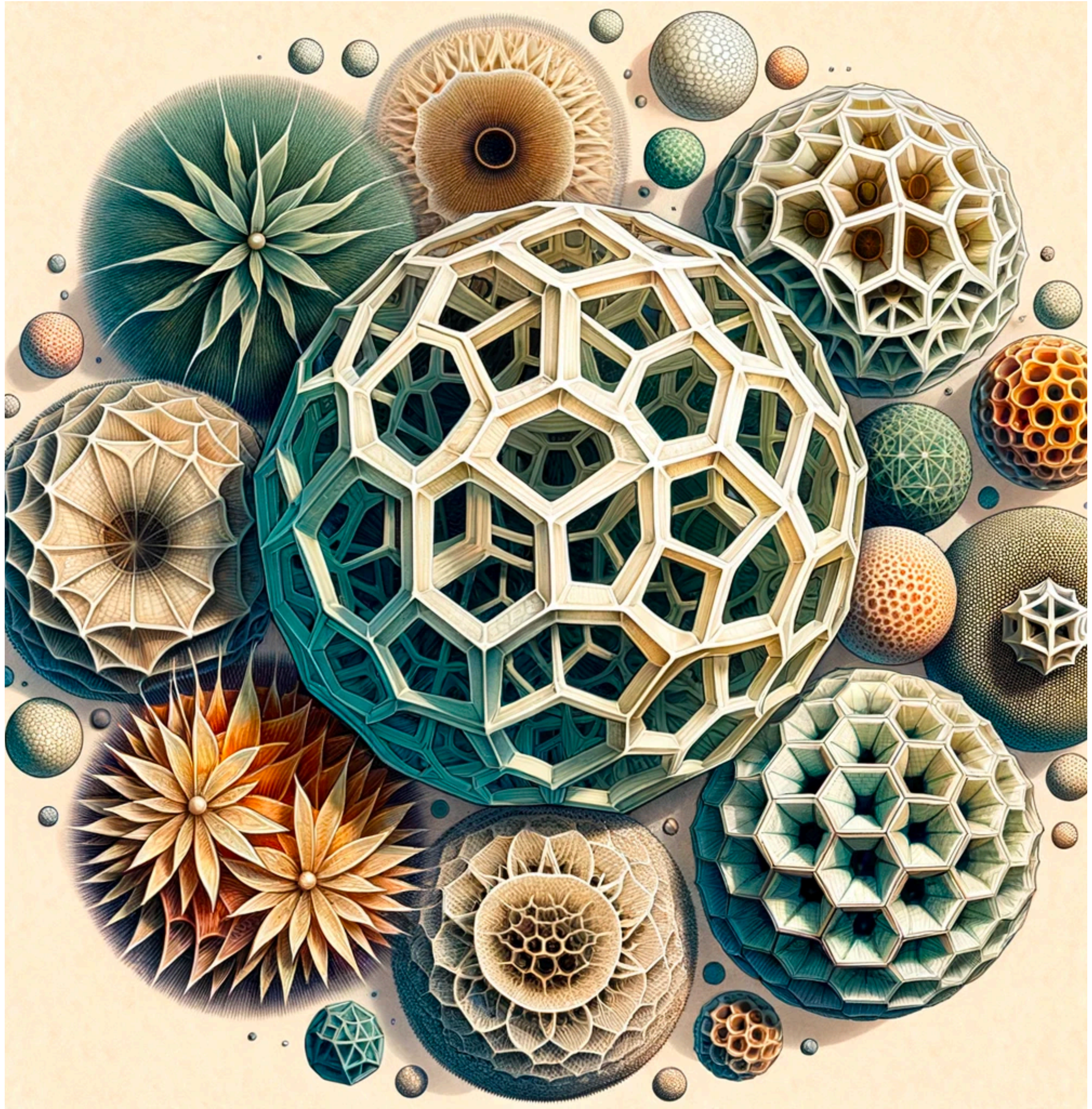
Algorithms as Phases

From Linear Logic to Singular Learning Theory

Daniel Murfet 12/11/23

"All of this will lead to theories which are much less rigidly of an all-or-nothing nature than past and present formal logic. They will be of a much less combinatorial, and much more analytical, character. In fact there are numerous indications to make us believe that this new system of formal logic will move closer to another discipline which has been little linked in the past with logic. This is thermodynamics, primarily in the form it was received from Boltzmann, and is that part of theoretical physics which comes nearest in some of its aspects to manipulating and measuring information."

John von Neumann, Collected Works, Vol. 5, p.304



Questions

Encountered in the "wild"

- Are Large Language Models (LLMs) reasoning?
- How is that reasoning represented at a computational level?
- How does that reasoning emerge during the training process?
- What kind of mathematical / statistical phenomena is that emergence?
- **What is the emergent logic of large scale learning machines?**

Outline

1. Proofs, Programs and Learning
2. Introduction to Singular Learning Theory
3. The Singular Learning Process
4. ~~From Linear Logic to Singular Learning Theory~~
5. ~~Algorithms as Phases~~

References

- T. Waring "Geometric perspectives on program synthesis and semantics" MSc University of Melbourne 2021.
- J. Clift, D. M., J. Wallbridge "Geometry of Program Synthesis" 2020.
- J. Clift and D. M. "Derivatives of Turing Machines in Linear Logic" 2018.
- The Gray Book, S. Watanabe "Algebraic Geometry and Statistical Learning Theory", 2009.
- The WBIC paper, S. Watanabe "A Widely Applicable Bayesian Information Criterion" JMLR 2013.
- The Green Book, S. Watanabe "Mathematical Theory of Bayesian Statistics", 2018.

Proofs, Programs and Learning

Proofs as Constructions

$$
\begin{prooftree}
\AxiomC{}
\UnaryInfC{$A \vdash A$}
\AxiomC{}
\UnaryInfC{$A \vdash A$}
\AxiomC{}
\UnaryInfC{$A \vdash A$}
\RightLabel{$\multimap L$}
\BinaryInfC{$A, A \multimap A \vdash A$}
\RightLabel{$\multimap L$}
\BinaryInfC{$A, A \multimap A, A \multimap A \vdash A$}
\RightLabel{$\multimap R$}
\UnaryInfC{$A \multimap A, A \multimap A \vdash A \multimap A$}
\RightLabel{der}
\UnaryInfC{$!(A \multimap A), A \multimap A \vdash A \multimap A$}
\RightLabel{der}
\UnaryInfC{$!(A \multimap A), !(A \multimap A) \vdash A \multimap A$}
\RightLabel{ctr}
\UnaryInfC{$!(A \multimap A) \vdash A \multimap A$}
\end{prooftree}
$$

Proofs, Programs and Learning

- In Nature many structures arise through learning processes
- In 1948 Turing introduced the idea of "unorganised machines" (essentially a kind of neural network) that could be driven towards organisation by interaction with data, and proposed this as a model of human development.
- Are trained neural networks "programs"?
- If we identify the structure of a proof with the structure of its construction, and view learning processes as "constructions", it leads us to ask about the **logical structure of learning processes**.

Introduction to Singular Learning Theory

Bayesian Statistics

- Bayesian statistics is about learners making observations of a generating process in the environment, and attempting to predict it. The basic ingredients are the *true distribution* $q(x)$, the class of models $p(x \mid w)$ with parameter $w \in W$ and *prior* $\varphi(w)$.
- The more samples $D_n = \{X_1, \ldots, X_n\}$ you see from the true distribution, the better able you are to find a model $p(x \mid w)$ which "fits" those samples.
- But it's not enough to just fit the data, since ultimately you want to *predict*.
- Bayesian statistics gives a powerful mathematical framework for reasoning about the **tradeoff** between explaining the data you have already seen, and predicting the data you are about to see.

Bayesian Statistics

- Basic ingredients: *true distribution* $q(x)$, the class of models $p(x \mid w)$ with parameter $w \in W$ and *prior* $\varphi(w)$ and samples $D_n = \{X_1, \ldots, X_n\}$ from the true distribution.
- $p(D_n \mid w) = \prod_{i=1}^{n} p(X_i \mid w)$ and $p(w \mid D_n)p(D_n) = p(w, D_n) = p(D_n \mid w)\varphi(w)$
- This yields a formula for the *Bayesian posterior* (belief after seeing data)

$$p(w \mid D_n) = \frac{1}{Z_n} p(D_n \mid w)\varphi(w)$$

where $Z_n = \int p(D_n \mid w)\varphi(w)dw$ is called the *partition function* or *marginal likelihood*.

- The *predictive distribution* is $p^*(x) = \int p(x \mid w)p(w \mid D_n)dw$. According to Bayesian statistics, this is how you "should" believe in future samples from the true distribution.

Bayesian Statistics

- Basic ingredients: *true distribution* $q(x)$, the class of models $p(x \mid w)$ with parameter $w \in W$ and *prior* $\varphi(w)$ and samples $D_n = \{X_1, \ldots, X_n\}$ from the true distribution. The Bayesian posterior $p(w \mid D_n) = \frac{1}{Z_n} p(D_n \mid w)\varphi(w)$.
- Recall $Z_n = p(D_n)$ is how likely this *model* thinks the data is. If you have another model $p'(x)$ you get $Z'_n = p'(D_n)$ and if that model thinks this data is more likely, that is, $Z'_n > Z_n$ then you "should" switch to that model.
- **Bayesian model selection**: prefer the model with the *highest* marginal likelihood Z_n or what is the same, the *lowest free energy* $F_n = -\log Z_n$.

Bayesian Statistics

- Basic ingredients: *true distribution* $q(x)$, the class of models $p(x \mid w)$ with parameter $w \in W$ and *prior* $\varphi(w)$ and samples $D_n = \{X_1, \ldots, X_n\}$ from the true distribution. The Bayesian posterior $p(w \mid D_n) = \frac{1}{Z_n} p(D_n \mid w)\varphi(w)$. Prefer the model with the lowest free energy $F_n = -\log Z_n$.
- **Classical Bayesian statistics (BIC):** for large n, $F_n \approx nL_0 + \frac{d}{2}\log n$ where L_0 is the negative log likelihood, think of it as the KL divergence between the "best" model and the true distribution, and d is the number of parameters (Schwarz).
- **Modern Bayesian statistics (WBIC):** for large n, $F_n \approx nL_0 + \lambda \log n$ where λ, the learning coefficient, may be less than $\frac{d}{2}$ (Watanabe).

Singular Learning Theory

- Basic ingredients: *true distribution* $q(x)$*,* the class of models $p(x \mid w)$ with parameter $w \in W$ and *prior* $\varphi(w)$ and samples $D_n = \{X_1, \ldots, X_n\}$. The Bayesian posterior is $p(w \mid D_n) = \frac{1}{Z_n} p(D_n \mid w)\varphi(w)$. For large n, $F_n \approx nL_0 + \lambda \log n$.
- Classical Bayesian statistics only applies to *regular models* (where the map from parameters w to models $p(x \mid w)$ is locally injective).
- Neural networks and other models with hidden variables are *singular* which means that $\lambda < \frac{d}{2}$.
- **Singular Learning Theory (SLT)** is a modern theory of Bayesian statistics, developed by Sumio Watanabe and collaborators, over the last twenty years which extends Bayesian statistics to singular models (using empirical process theory, functional analysis, algebraic geometry).
- SLT is one of the leading candidates for a mathematical theory of deep learning.

Singular Learning Process

Algorithms as endpoints of Learning

- To apply Bayesian statistics we need a generating process, or true distribution.
- Suppose we have for each $x \in X$ a corresponding proof $x : A$ and for each $y \in Y$ a proof $y : B$ and let $f : X \to Y$ be a given function. We take pairs (x, y) with $y = f(x) + \varepsilon$ as samples from our true distribution and ask: **which algorithm produced these samples**?
- We can imagine a learning process which starts with "confusion" and ends with an algorithm for computing f, in such a way that the structure of the learning process reflects something about the structure of the algorithm.
- **Questions**: how to set up a "space" W of algorithms in LL? What is the model? What kind of structure do learning processes have? What is structure of algorithms?

Singular Learning Process

Gray Book, Section 7.6.

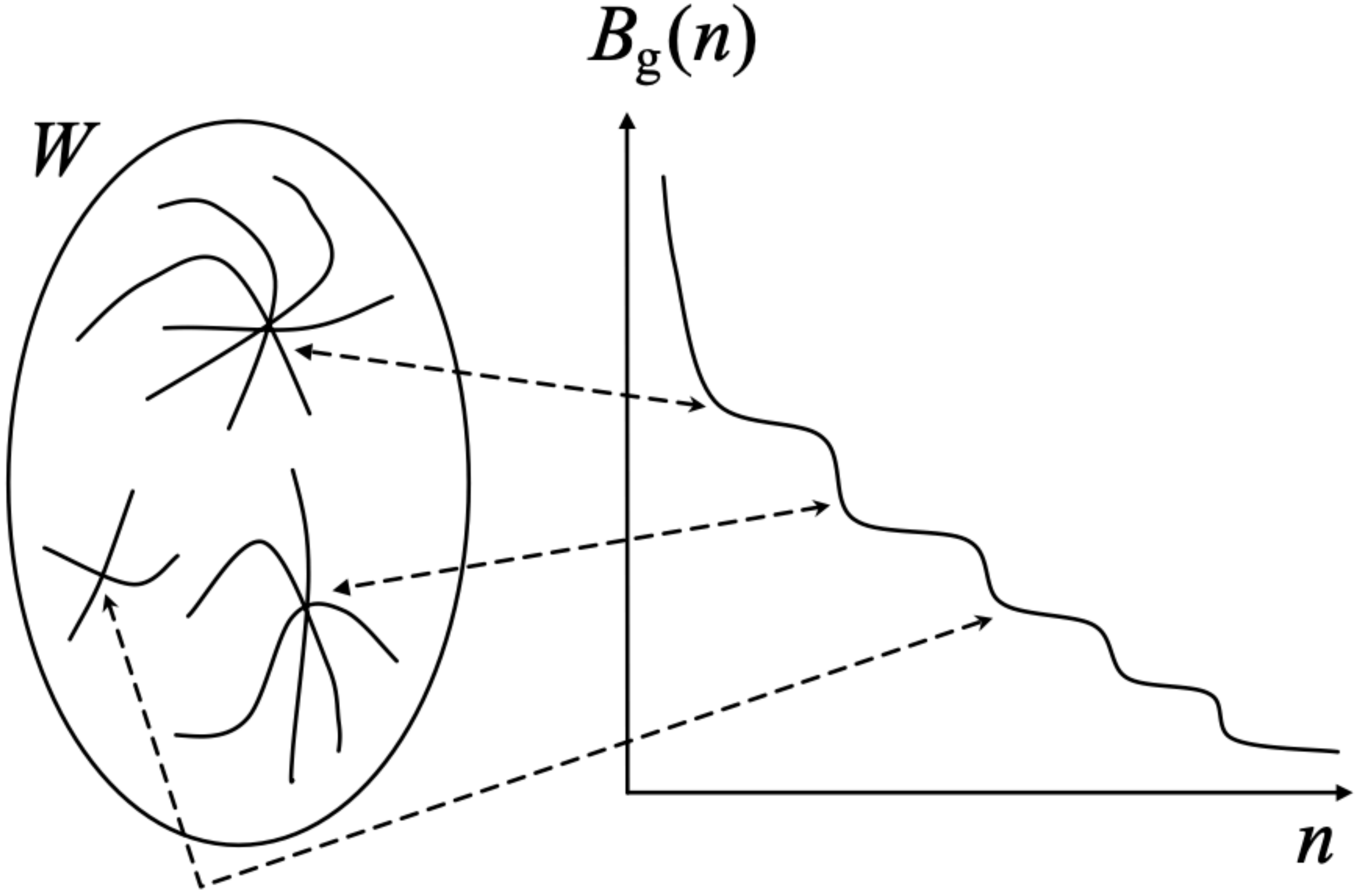


Fig. 7.6. Learning curve with singularities

Setup

- Samples $X_1, \ldots, X_n$ are independently subject to a true distribution $q(x)$. We denote by $p(x \mid w)$ our model and $\varphi(w)$ our prior, on parameter space W.
- The negative log likelihood is $L_n(w) = -\frac{1}{n}\sum_{i=1}^{n} \log p(X_i \mid w)$
- The (Bayes) free energy is defined to be

$$
\begin{aligned}
F_n &= -\log \int \prod_{i=1}^{n} p(X_i \mid w)\varphi(w)dw \\
&= -\log \int \exp(-nL_n(w))\varphi(w)dw
\end{aligned}
$$

Setup

For Neural Networks

- The true distribution is $q(x, y) = q(y \mid x)q(x)$ with inputs $x \in \mathbb{R}^m$ and outputs $y \in \mathbb{R}^n$. We denote by $p(x, y \mid w) = p(y \mid x, w)q(x)$ our model and $\varphi(w)$ our prior, on parameter space W. Suppose given samples $(X_1, Y_1), \ldots, (X_n, Y_n)$.
- The model is given by $p(y \mid x, w) = \dfrac{1}{(2\pi)^{n/2}} \exp\left(-\dfrac{1}{2} \parallel y - f(x, w) \parallel^2\right)$ where $f(x, w)$ is a neural network with weights w.
- In this case the log loss is the mean squared error (up to some constants).

Setup

For Neural Networks

$$
\begin{aligned}
L_n(w) &= -\frac{1}{n}\sum_{i=1}^{n} \log p(X_i, Y_i \mid w) \\
&= -\frac{1}{n}\sum_{i=1}^{n} \log\Big[\frac{1}{(2\pi)^{n/2}} \exp\Big(-\frac{1}{2}\ \|\ Y_i - f(X_i, w)\ \|\ ^2\Big) q(X_i)\Big] \\
&= \frac{1}{2n}\sum_{i=1}^{n}\ \|\ Y_i - f(X_i, w)\ \|\ ^2 - \frac{1}{n}\sum_{i=1}^{n} \log q(X_i) + \text{const.}
\end{aligned}
$$

Mean squared error, i.e. "loss"

Empirical entropy of $q(x)$

Setup

For Neural Networks

$$L_n(w) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{2}\,\| \, Y_i - f(X_i, w) \, \|^2 - \frac{1}{n}\sum_{i=1}^{n}\log q(X_i) + \text{const.}$$

$$F_n = -\log \int \prod_{i=1}^{n} p(X_i \,|\, w)\varphi(w)dw$$

$$= -\log \int \exp(-nL_n(w))\varphi(w)dw$$

$$= -\log Z_n$$

Partition function / model evidence

$$Z_n = \int \exp(-nL_n(w))\varphi(w)dw$$

Bayesian posterior

$$p(w \,|\, D_n) = \frac{1}{Z_n}\exp(-nL_n(w))\varphi(w)$$

Free Energy Formula

Precise Statement

- Assume *relative finite variance* [**Green**, §3.1] in addition to the fundamental conditions of [**Gray**] (excepting realisability) and that there is a point w_0 minimising L in the interior of W.

- **Theorem** (Watanabe): We have by [**Green**, §6.3], see also [**WBIC, Renormalizability**]:

$$F_n = nL_n(w_0) + \lambda \log n - (m-1)\log\log n + F_n^R + o_p(1)$$

- Here $\lambda \in \mathbb{Q}_{>0}$ is called the *learning coefficient*, $m \in \mathbb{N}$ is the *multiplicity* and F_n^R is a random variable which converges to a random variable in law.

Internal Model Selection

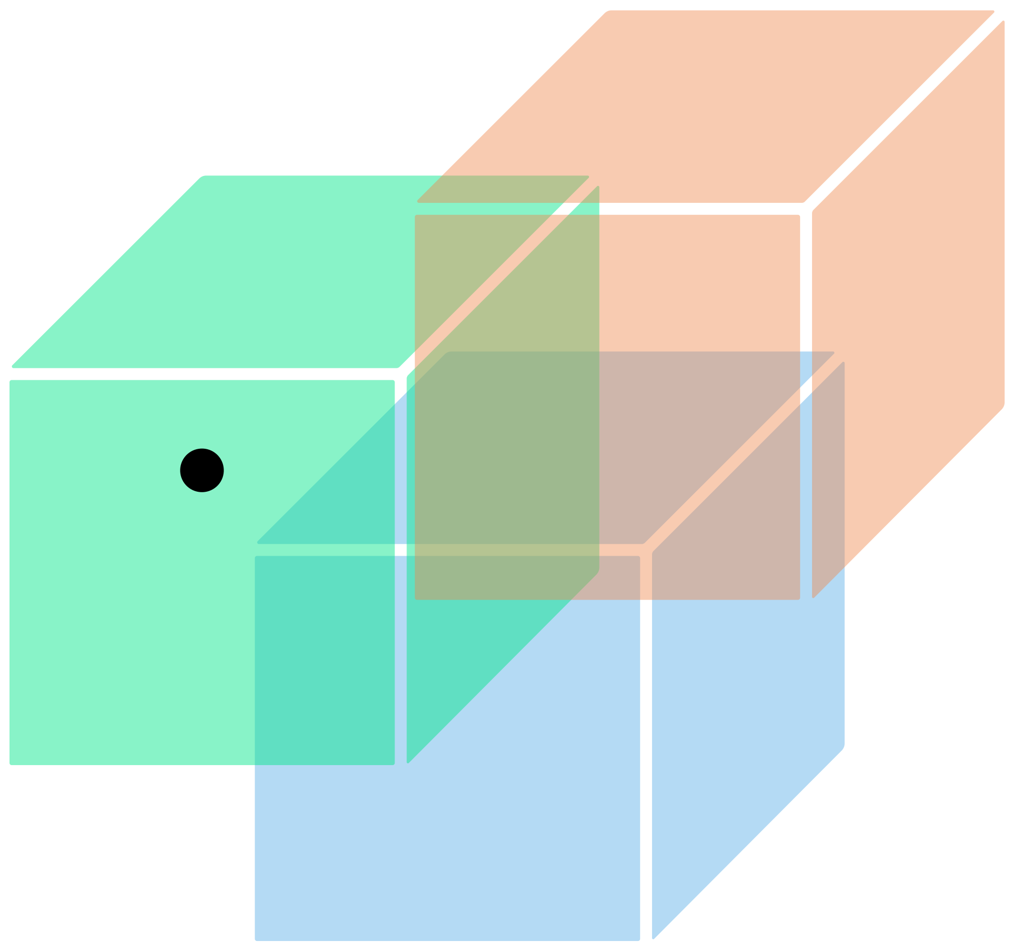
- Model selection is usually thought of something that statisticians do.
- Nontrivial prediction of SLT: model selection can happen **automatically** in Bayesian learning, **internally** to a single model.
- Given a model (p, q, φ) with parameter space W we refer to the emergent submodels W_α, between which this internal model selection chooses, as *phases*. A change in n leading to a different choice is called a *phase transition*.
- For clarity we sometimes call this a *Bayesian phase transition*.

Internal Model Selection

$$F_n = -\log \int_W e^{-nL_n(w)} \varphi(w) dw$$

$$= -\log \sum_\alpha \int_{W_\alpha} e^{-nL_n(w)} \varphi_\alpha(w) dw$$

$$= -\log \sum_\alpha e^{-F_n(W_\alpha)}$$



- Here $F_n(W_\alpha) = -\log \int_{W_\alpha} exp(-nL_n(w))\varphi_\alpha(w)dw$ is (essentially) the free energy of the submodel with parameter space W_α, prior $\varphi'_\alpha = \frac{1}{V_\alpha}\varphi_\alpha$ where $V_\alpha = \int_{W_\alpha} \varphi_\alpha$, and the same model p, truth q as the original.

Internal Model Selection

- We can apply the Free Energy Formula to the model $(p, q, \varphi'_\alpha, W_\alpha)$ to obtain

$$F_n(W_\alpha) \approx nL_n(w^*_\alpha) + \lambda_\alpha \log n + c_\alpha$$

- Then

$$F_n = -\log \sum_\alpha e^{-F_n(W_\alpha)} \approx \min_\alpha F_n(W_\alpha)$$

$$\approx \min_\alpha \left[nL_n(w^*_\alpha) + \lambda_\alpha \log n + c_\alpha \right]$$

- The Bayesian posterior **selects** phases on the basis of competition between *energy, complexity* and subleading terms (which include prior effects). When the index α changes as a function of n or hyperparameters, we say that there has been a *phase transition* in the Bayesian posterior.

Thermodynamics

- Now we take the **Free Energy Formula** and the principle of **Internal Model Selection** and do "thermodynamics" that is, we deduce several interesting facts about learning machines from elementary manipulations of the formula

$$F_n = -\log \sum_{\alpha} e^{-F_n(W_\alpha)} \approx \min_{\alpha} F_n(W_\alpha)$$

$$\approx \min_{\alpha} \left[nL_n(w_\alpha^*) + \lambda_\alpha \log n + c_\alpha \right]$$

- To start with make two additional simplifying assumptions: replacing $L_n(w_\alpha^*)$ by the deterministic $L_\alpha := L(w_\alpha^*)$ and assuming that $c_\alpha = 0$.

Thermodynamics

- If a phase α is dominated by a phase β both with respect to energy $L_\alpha > L_\beta$ and learning coefficient $\lambda_\alpha > \lambda_\beta$ then $F_n(W_\alpha) > F_n(W_\beta)$ but there is **no phase transition** because this is true for all n.
- For there to be a phase transition in n between phases $\alpha \longrightarrow \beta$ we need both a *critical dataset size* $n = n_{cr}$ and for this transition to not be "screened" by others:

$F_n(W_\alpha) < F_n(W_\beta)$ $\quad$ $F_{n_{cr}}(W_\alpha) \approx F_{n_{cr}}(W_\beta)$ $\quad$ $F_n(W_\alpha) > F_n(W_\beta)$

Thermodynamics

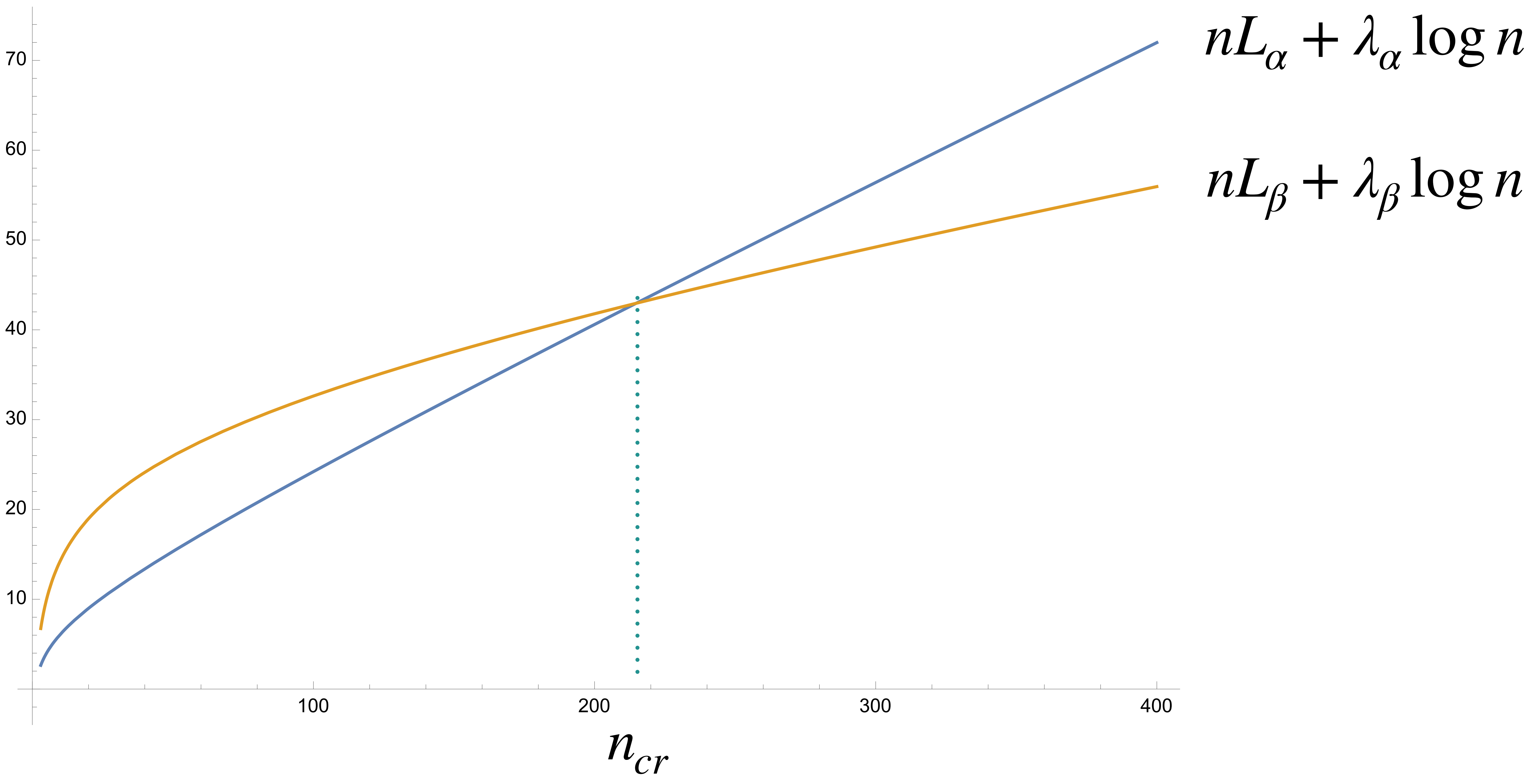
- Assume without loss of generality that $L_\alpha > L_\beta$ and $\lambda_\alpha < \lambda_\beta$. Then

$$
\begin{aligned}
F_n(W_\alpha) = F_n(W_\beta) \quad &\iff \quad nL_\alpha + \lambda_\alpha \log n = nL_\beta + \lambda_\beta \log n \\
&\iff \quad n(L_\alpha - L_\beta) = -\log n(\lambda_\alpha - \lambda_\beta) \\
&\iff \quad \frac{n}{\log n} = -\frac{\lambda_\beta - \lambda_\alpha}{L_\beta - L_\alpha} = -\frac{\Delta\lambda}{\Delta L}
\end{aligned}
$$

- The function $n/\log n$ is positive and increasing for $n > e$ so this has a unique solution, which is the critical dataset size n_{cr}.

Thermodynamics

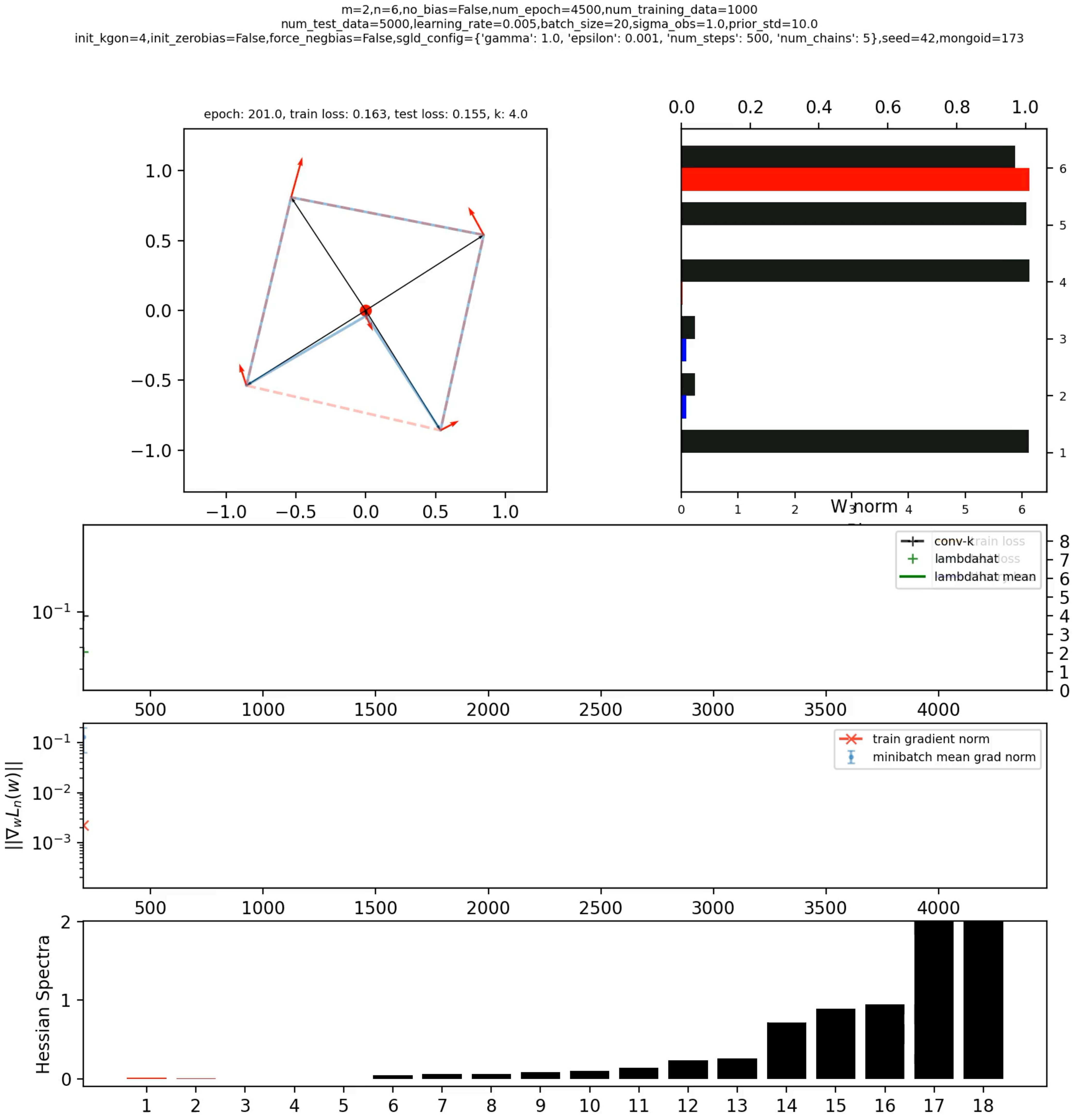
- If $L_\alpha > L_\beta$ and $\lambda_\alpha < \lambda_\beta$ then there is a (candidate) transition $\alpha \longrightarrow \beta$



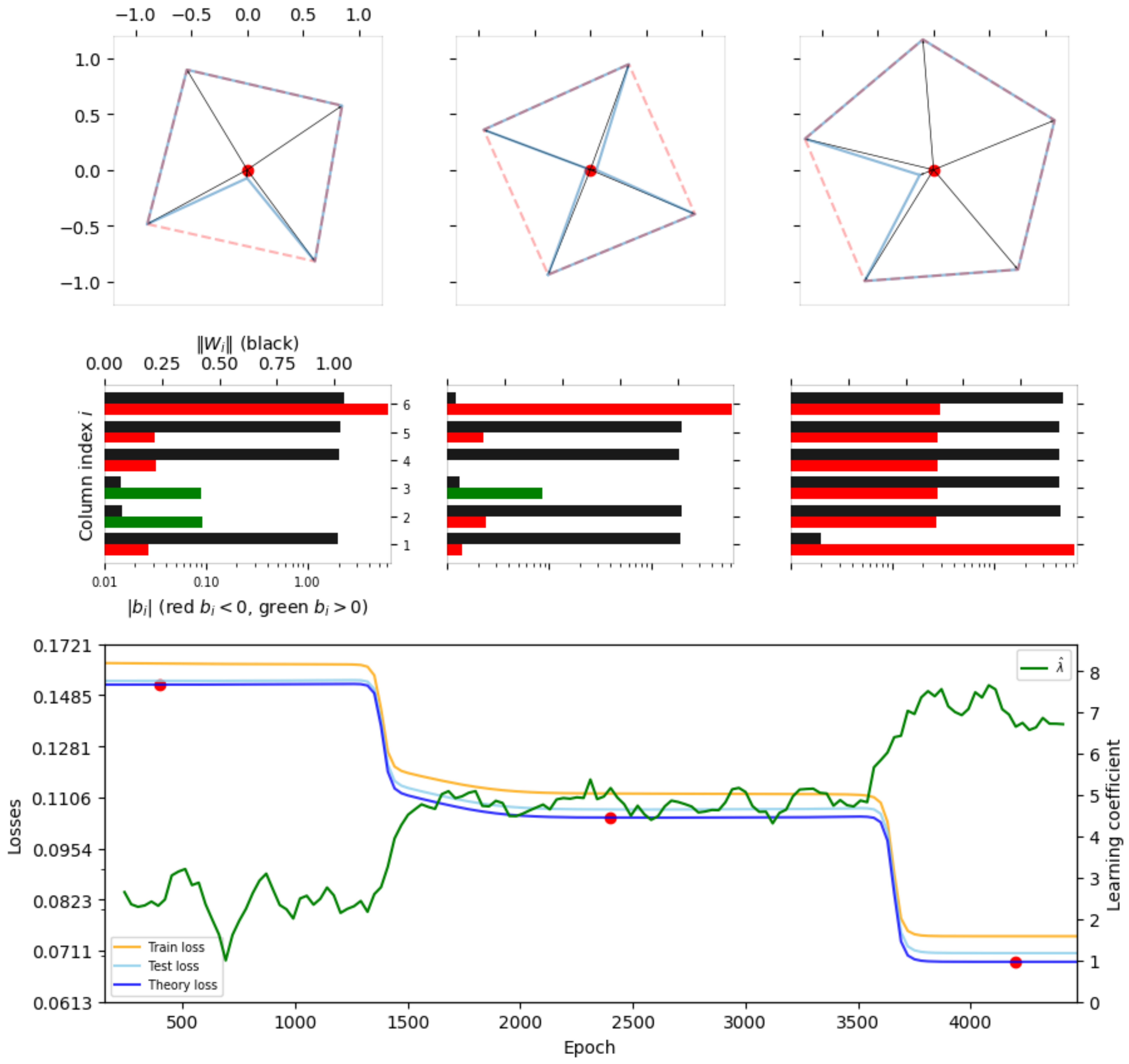
Thermodynamics

- Assuming that $L_\alpha > L_\beta$ and $\lambda_\alpha < \lambda_\beta$ there is a (candidate) phase transition in the Bayesian posterior $\alpha \longrightarrow \beta$ at $n = n_{cr}$. We call this the *critical dataset size* for the transition.
- **Type A.** Phase transitions in n that change the energy must *decrease* the energy and *increase* the learning coefficient.

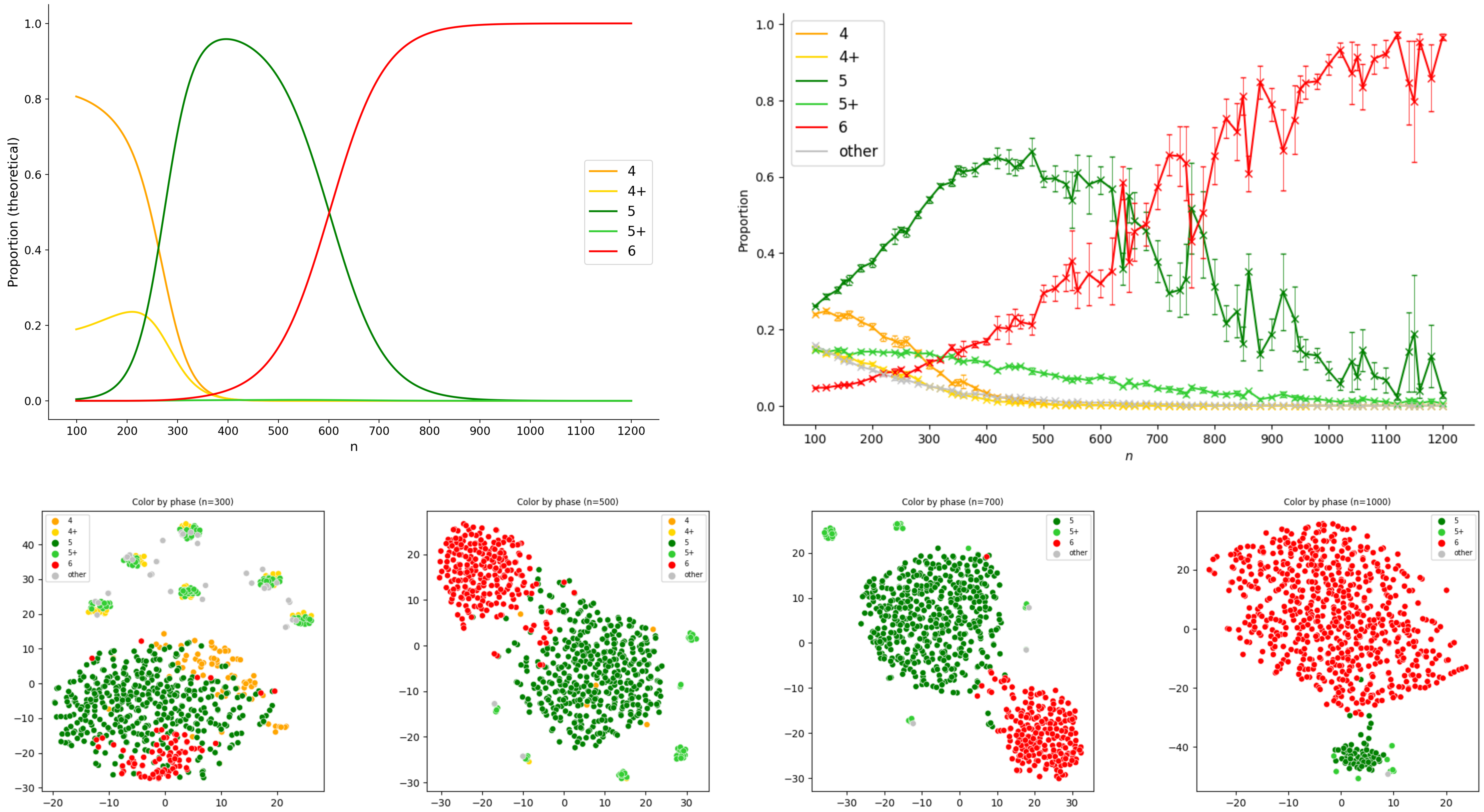
"The learning process produces *more accurate* models that are *more complex,* sacrificing extra bits in the model description for fewer errors"



"Dynamical versus Bayesian Phase Transitions in a Toy Model of Superposition" Z. Chen, E. Lau, J. Mendel, S. Wei, D. M, arXiv: 2310.06301.



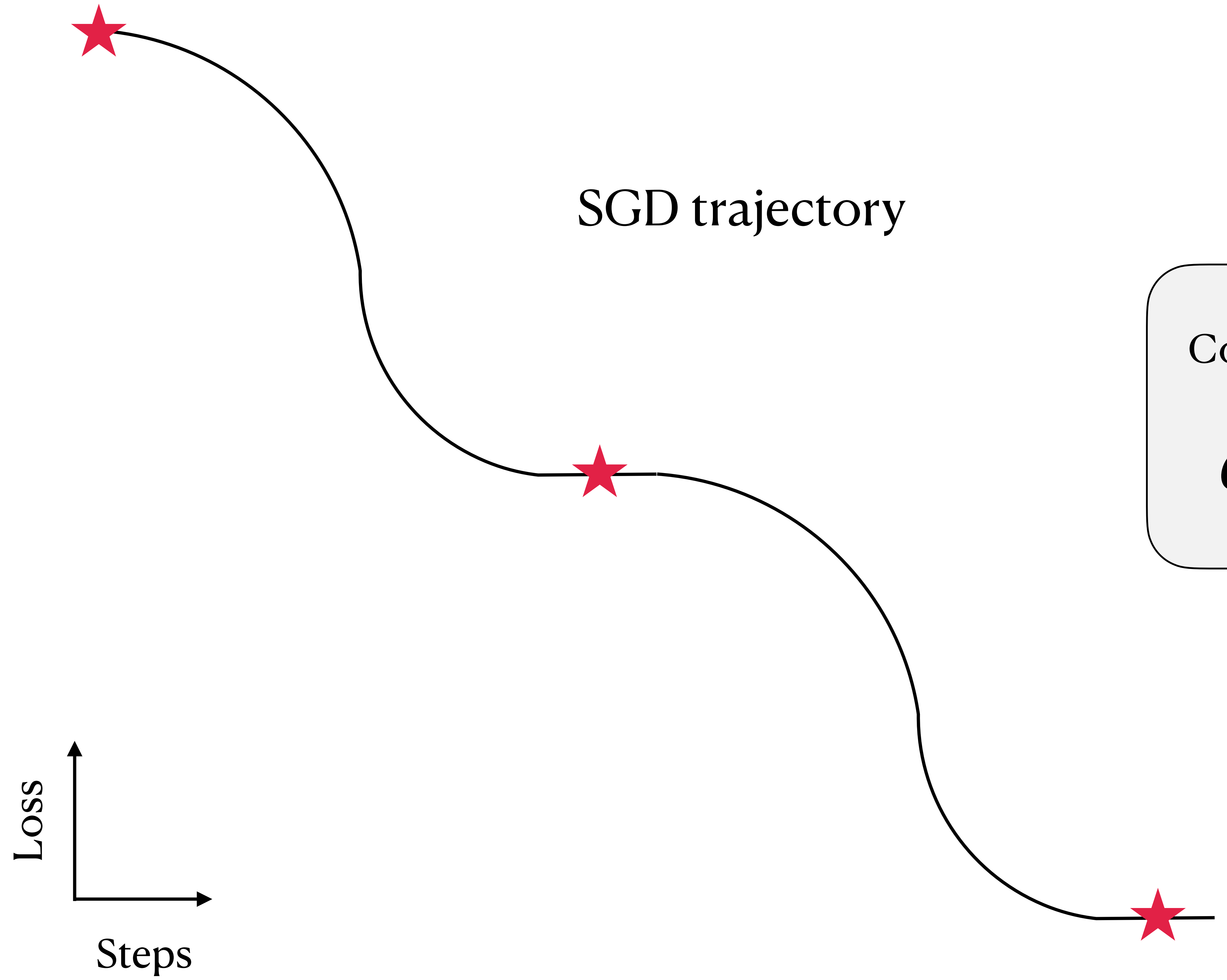
−1.0 −0.5 0.0 0.5 1.0
1.0
0.5
0.0
−0.5
−1.0
$\|W_i\|$ (black)
0.00 0.25 0.50 0.75 1.00
Column index i
6
5
4
3
2
1
0.01 0.10 1.00
$|b_i|$ (red $b_i < 0$, green $b_i > 0$)
0.1721
0.1485
0.1281
0.1106
0.0954
0.0823
0.0711
0.0613
Losses
500 1000 1500 2000 2500 3000 3500 4000
Epoch
$\hat{\lambda}$
8 7 6 5 4 3 2 1 0
Learning coefficient
Train loss
Test loss
Theory loss



Proportion (theoretical)
n
4
4+
5
5+
6
Proportion
other
Color by phase (n=300)
Color by phase (n=500)
Color by phase (n=700)
Color by phase (n=1000)

SGD vs Bayes

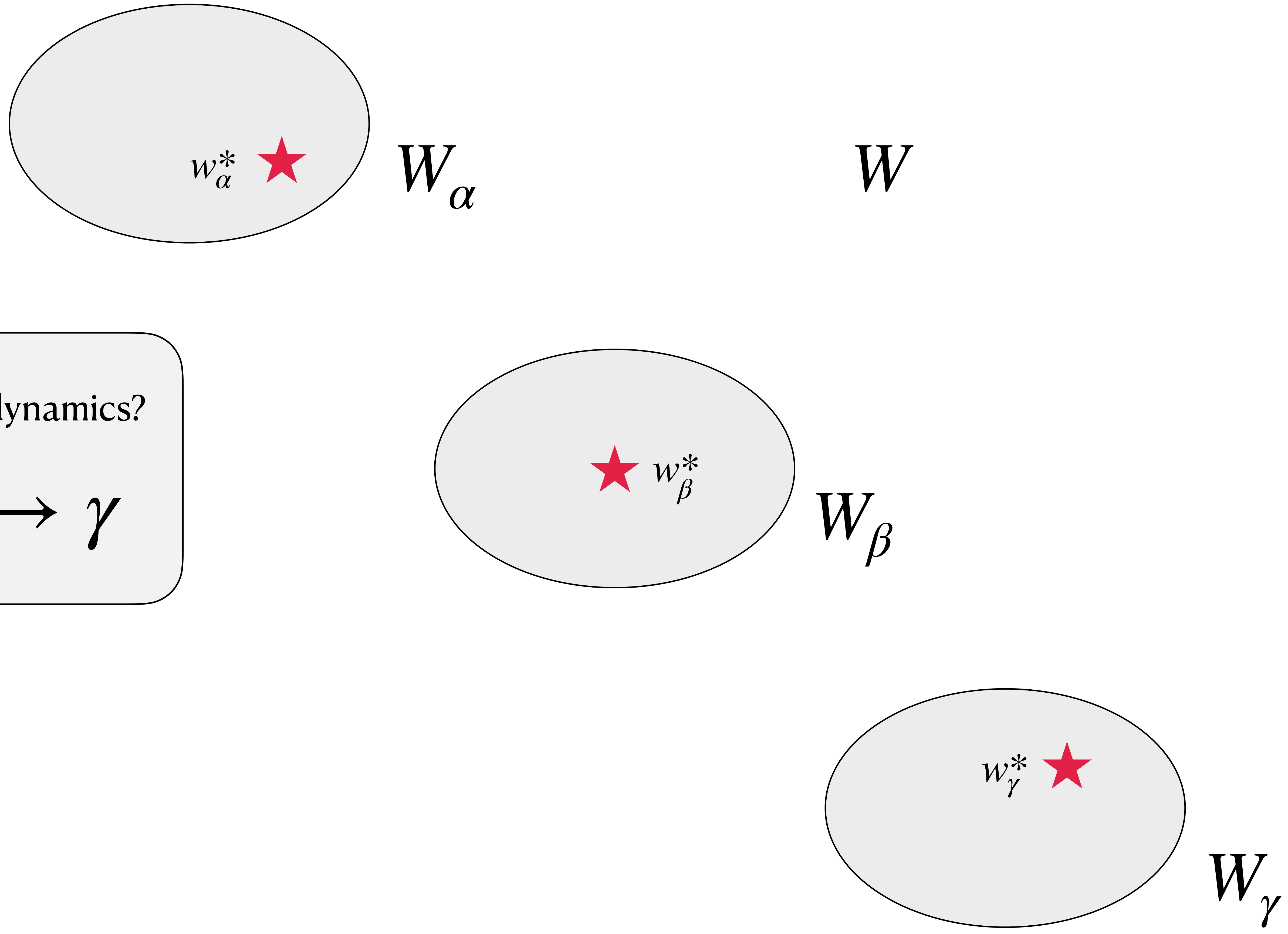
Coarse-grained



Dynamical Transition

Coarse-grained dynamics?

$$\alpha \rightarrow \beta \rightarrow \gamma$$



Bayesian Transition

Structure vs Structure

- Associate to a sequent $\Gamma \vdash B$ in linear logic and constraints (e.g. input-output behaviour) a learning problem in SLT such that **local structure** of the learning process near a (partial) solution $\pi : \Gamma \vdash B$ reflects **structure** of π.
- The kind of structure that we expect to be visible includes "degeneracy", "symmetry", "factorisation", "modularity" but we lack examples.
- If we have a robust mathematical theory of the singular learning process in a logical setting where we independently understand what "structure" means, it might give us hints about how to build a mathematical theory of emergent logic in other learning machines.

Geometry of Program Synthesis

- Structure of learning processes in SLT
- $\leftrightarrow$ structure of singularities
- $\leftrightarrow$ equations among derivatives of negative log likelihood L
- $\leftrightarrow$ differential equations in differential linear logic
- $\leftrightarrow$ structure of algorithms