Modeling the Dunning–Kruger Effect within the Lindian Levels of Functionality Framework

Jason L. Lind

April 17, 2025

Abstract

The Dunning–Kruger effect describes a cognitive bias whereby individuals with lower ability overestimate their competence, while those with higher ability may underestimate theirs. This paper integrates the Dunning–Kruger phenomenon into the Lindian Levels of Functionality framework, which models actors' population (o) and aggregate influence (Φ) across a continuum of development stages from Programming to Transcendence. We formalize perceived versus actual influence dynamics, propose a generative model for cognitive bias, and introduce a population-weighted organizational metric. Finally, we interpret these dynamics specifically regarding interactions and perceptions associated with the use of Large Language Models (LLMs).

1 Introduction

The Dunning–Kruger effect reveals an inherent asymmetry in human self-assessment, leading to systemic over- and under-confidence based on true competence levels. When mapped into the Lindian levels of functionality (ω_1 to ω_5), this bias can be analyzed using scalar fields for actor populations and their aggregate influence. This mapping provides insights into cognitive dynamics within organizations, particularly in understanding differing approaches and biases in interacting with and perceiving Large Language Models (LLMs).

2 Recall – Lindian Scalar Fields

Let $\omega \in [1, 5]$ represent continuous levels of functionality. We define:

$$o(\omega) = \int_{x=\omega}^{\omega+1} \omega_x \, d(-\infty) \qquad \text{(actor population at level } \omega\text{)} \tag{1}$$

$$\Phi(\omega) = \iint_{x=\omega}^{\omega+1} \omega_x \, d(-\infty) \, d(+\infty) \qquad \text{(aggregate influence at level } \omega\text{)} \tag{2}$$

3 Perceived vs. Actual Influence in LLM Interactions

Define:

- $\Phi(\omega)$: actual aggregate influence resulting from interactions with LLMs
- $\Psi(\omega)$: perceived self-assessed influence and competence in using LLMs

The mis-estimation (bias) is:

$$\Delta(\omega) = \frac{\Psi(\omega) - \Phi(\omega)}{\Phi(\omega)} \tag{3}$$

4 Generative Bias Model for LLM Usage

Novices interacting with LLMs extrapolate from their limited experience. We model this with a first-order Taylor expansion of Φ toward the maximum level $\omega_{\max} = 5$:

$$\Psi(\omega) \approx \Phi(\omega) + (\omega_{\max} - \omega)\frac{\partial \Phi}{\partial \omega}. \tag{4}$$

Thus, the bias becomes:

$$\boxed{\Delta(\omega) = \frac{(\omega_{\max} - \omega)\frac{\partial \Phi}{\partial \omega}}{\Phi(\omega)}}, \quad (\omega_{\max} = 5) \tag{5}$$

This captures:

- **Low-skill ($\omega \approx 1$):** Over-confidence in simple prompt engineering
- **Mid-skill ($\omega \approx 3$):** Under-confidence due to recognition of LLM limitations
- **High-skill ($\omega \to 5$):** Calibrated confidence through deep understanding and meta-modeling of LLM capabilities

5 Population-Weighted Dunning–Kruger Score

To assess LLM adoption across an organization, weight by actor population $o(\omega)$:

$$\text{DK\_score} = \frac{\int_1^5 \Delta(\omega)\, o(\omega)\, d\omega}{\int_1^5 o(\omega)\, d\omega}. \tag{6}$$

This metric provides an organizational measure of cognitive calibration specifically for LLM usage.

6 Application to Organizational LLM Integration

In LLM-integrated environments:

- ω_1–ω_2 users (programmers, tool builders) often overestimate their prompt engineering skill.
- ω_3 engineers, more familiar with complexities, tend toward under-confidence.
- ω_4–ω_5 transcenders develop sophisticated methodologies and frameworks that align perceived influence closely with actual effectiveness.

Instrumentation involves tracking objective performance metrics and self-assessment data to monitor and correct misalignments over time.

7 Conclusion

The Lindian Levels of Functionality framework enables formal modeling of the Dunning–Kruger effect specifically in the context of LLM interactions. By clearly distinguishing between perceived and actual influence, organizations can strategically manage cognitive biases during rapid technological adoption phases. Future work includes more nuanced models incorporating second-order dynamics and adaptive feedback mechanisms.

Unweighted Jacobian Matrix

$$\mathbf{J}_{new} = \begin{bmatrix} \alpha\dfrac{\partial P}{\partial S} + \beta\dfrac{\partial S(I,t)}{\partial S} + D & \alpha\dfrac{\partial P}{\partial I} + \beta\dfrac{\partial S(I,t)}{\partial I} & \alpha\dfrac{\partial P}{\partial T} \\ \gamma\dfrac{\partial (S-I)}{\partial S} + \nabla\cdot\left(\frac{\partial g}{\partial S}\,\nabla I\right) & -\delta + \gamma\dfrac{\partial (S-I)}{\partial I} + \nabla\cdot\left(\frac{\partial g}{\partial C}\,\nabla I\right) & \gamma\dfrac{\partial (S-I)}{\partial T} \\ \dfrac{\partial h}{\partial S} & \dfrac{\partial h}{\partial I} & -\zeta + \delta\left(1 - \frac{T}{S}\right) + \delta\,\dfrac{\partial\left(1 - T/S\right)}{\partial T} \end{bmatrix}$$

Key Jacobian Entries

1. $\mathbf{A} = \alpha\dfrac{\partial P}{\partial S} + \beta\dfrac{\partial S(I,t)}{\partial S} + D$
   Mixes how probabilistic effects and structured influence feed back into **Space** (plus any diffusion term).

2. $\mathbf{B} = \alpha\dfrac{\partial P}{\partial I} + \beta\dfrac{\partial S(I,t)}{\partial I}$
   Captures how both probability and structure impact **Thought**.

3. $\mathbf{C} = \alpha\dfrac{\partial P}{\partial T}$
   Measures how probabilistic dynamics project into **Time**.

4. $\mathbf{D_1} = \gamma \dfrac{\partial (S-I)}{\partial S} + \nabla \cdot \big((\partial g / \partial S)\, \nabla I \big)$
Blends the logistic "gap" between Space and Thought with diffusion of influence gradients back into **Space**.

5. $\mathbf{E} = -\delta + \gamma \dfrac{\partial (S-I)}{\partial I} + \nabla \cdot \big((\partial g / \partial C)\, \nabla I \big)$
Self-decay of perceived influence, plus how Space–Thought gaps and cyber-driven diffusion shape **Thought**.

6. $\mathbf{F} = \gamma \dfrac{\partial (S-I)}{\partial T}$
How the Space vs. Thought "distance" changes over **Time**.

7. $\mathbf{G} = \dfrac{\partial h}{\partial S}$
Sensitivity of the transformation dynamics to **Space**.

8. $\mathbf{H} = \dfrac{\partial h}{\partial I}$
Sensitivity of the transformation dynamics to **Thought**.

9. $\mathbf{I_1} = -\zeta + \delta\,(1 - T/S) + \delta\, \dfrac{\partial (1 - T/S)}{\partial T}$
Base decay in transformation, plus how the current Time-to-Space ratio and its **Time**-derivative feed back.

$$J_{\text{new}} = \begin{bmatrix} A & B & C \\ D1 & E & F \\ G & H & I1 \end{bmatrix}.$$

Level ω_n	off-diag weight
ω_1 Programming	$\varepsilon = 0.05$
ω_2 Development	$\bullet = 0.25$
ω_3 Engineering	$\diamond = 0.60$
ω_4 Transformation	mix: C→S $\diamond$, C→T $\triangle$, all others $\triangle$
ω_5 Transcend	$\triangle = 1.00$

$$J_1 = \begin{bmatrix} 1 \cdot A & 0.05\,B & 0.05\,C \\ 0.05\,D1 & 1 \cdot E & 0.05\,F \\ 0.05\,G & 0.05\,H & 1 \cdot I1 \end{bmatrix}$$

$$J_2 = \begin{bmatrix} 1 \cdot A & 0.25\,B & 0.25\,C \\ 0.25\,D1 & 1 \cdot E & 0.25\,F \\ 0.25\,G & 0.25\,H & 1 \cdot I1 \end{bmatrix}$$

$$J_3 = \begin{bmatrix} 1 \cdot A & 0.60\,B & 0.60\,C \\ 0.60\,D1 & 1 \cdot E & 0.60\,F \\ 0.60\,G & 0.60\,H & 1 \cdot I1 \end{bmatrix}$$

$$J_4 = \begin{bmatrix} 1 \cdot A & 0.60\,B & 1.00\,C \\ 1.00\,D1 & 1 \cdot E & 1.00\,F \\ 1.00\,G & 1.00\,H & 1 \cdot I1 \end{bmatrix}$$

$$J_5 = \begin{bmatrix} 1 \cdot A & 1 \cdot B & 1 \cdot C \\ 1 \cdot D1 & 1 \cdot E & 1 \cdot F \\ 1 \cdot G & 1 \cdot H & 1 \cdot I1 \end{bmatrix}$$

These five matrices $\mathbf{J}_1 \dots \mathbf{J}_5$ explicitly show how the same core Jacobian update gets "relativistically" re-weighted at each level of functionality.

Here's a family of *closed-form, "textbook" analytic solutions* you can use as starting points at each level. In each case we approximate the weighted Jacobian by

$$J_n \;\approx\; a_n\, I \;+\; b_n\,\big(\mathbf{1}\mathbf{1}^T - I\big)$$

where

- a_n is the (level-specific) average of the three diagonal entries $\{A, E, I_1\}$,
- b_n is the (level-specific) off-diagonal scale factor times the average of the six off-diagonal symbols $\{B, C, D_1, F, G, H\}$,
- $\mathbf{1} = (1,1,1)^T$, and
- I is the 3×3 identity matrix.

This "uniform-coupling" approximation makes the spectrum, and hence the solution, **fully explicit**:

1. **Eigenvalues**

   $$\lambda_1 = a_n + 2\,b_n, \qquad \lambda_{2,3} = a_n - b_n,$$

   with eigenvectors

   $$v_1 = (1,1,1)^T, \quad v_2 = (1,-1,0)^T, \quad v_3 = (1,0,-1)^T.$$

2. **General solution** for the state $x(t) \in \mathbb{R}^3$ of $\dot{x} = J_n\, x$:

   $$x(t) = C_1\, v_1\, e^{\lambda_1 t} + C_2\, v_2\, e^{\lambda_2 t} + C_3\, v_3\, e^{\lambda_2 t},$$

   where the constants C_i are fixed by your initial condition $x(0)$.

Level-by-Level Parameters

Level	Off-diag weight b_n	Diag avg a_n ($\approx$)	Eigenvalues	Behavior
ω_1 Programming	$\varepsilon = 0.05$	$\bar{A}$	$\bar{A} + 0.10,\ \bar{A} - 0.05$	Almost pure $\exp(A\,t)$; ti
ω_2 Development	$\bullet = 0.25$	$\bar{A}$	$\bar{A} + 0.50,\ \bar{A} - 0.25$	Noticeable slower/fast m
ω_3 Engineering	$\diamond = 0.60$	$\bar{A}$	$\bar{A} + 1.20,\ \bar{A} - 0.60$	Strong "global" mode pl
ω_4 Transformation	mixed ($b \approx 0.80$)	$\bar{A}$	$\bar{A} + 1.60,\ \bar{A} - 0.80$	Very rapid "all-dims" bu
ω_5 Transcend	$\triangle = 1.00$	$\bar{A}$	$\bar{A} + 2.00,\ \bar{A} - 1.00$	Maximal leverage: one e

Here $\bar{A}$ denotes the arithmetic mean of $\{A, E, I_1\}$ for that level, and b_n multiplies the mean of the six off-diagonal symbols.

How to plug in your actual symbols

1. **Compute**

$$\bar{A} = \tfrac{1}{3}(A + E + I_1), \quad \bar{B} = \tfrac{1}{6}(B + C + D_1 + F + G + H).$$

2. **Set**

$$b_n = (\text{level-weight}) \times \bar{B}, \quad a_n = \bar{A}.$$

3. **Form** the approximate

$$J_n = a_n\, I + b_n\, (\mathbf{1}\mathbf{1}^T - I).$$

4. **Write** the solution

$$x(t) = C_1\,(1,1,1)^T\, e^{(a_n+2b_n)t} + C_2\,(1,-1,0)^T\, e^{(a_n-b_n)t} + C_3\,(1,0,-1)^T\, e^{(a_n-b_n)t}.$$

Why this matters:

- At ω_1, b_n is so tiny that all three modes collapse to nearly the same exponential $e^{a_1 t}$.
- As you climb levels, b_n grows, splitting the dynamics into one *fast "global"* expansion mode and two *slower "differential"* modes.

1 Starting points

Dunning–Kruger bias at functionality level $\omega \in \{1, \dots, 5\}$

$$\Delta(\omega) = \frac{\Psi(\omega) - \Phi(\omega)}{\Phi(\omega)} = \frac{(5-\omega)\,\partial_\omega \Phi}{\Phi(\omega)} \tag{DK–1}$$

Baseline level–weighted Jacobian family

$$J_n \;=\; a_n I \;+\; b_n\big(\mathbf{1}\mathbf{1}^\top - I\big), \qquad b_n = w_n\,\bar{B},\; a_n = \bar{A}, \tag{J–1}$$

$$w_1 = 0.05,\; w_2 = 0.25,\; w_3 = 0.60,\; w_4 \approx 0.80,\; w_5 = 1.00.$$

2 Bias–adjusted scaling

Introduce two hyper–parameters κ (off–diagonal sensitivity) and λ (diagonal sensitivity):

$$\boxed{\tilde{b}_n \;=\; b_n\big(1+\kappa\,\Delta(\omega_n)\big), \quad \tilde{a}_n \;=\; a_n + \lambda\,\Delta(\omega_n)} \tag{INT–1}$$

3 Integrated Jacobian

$$\boxed{\tilde{J}_n \;=\; \tilde{a}_n I \;+\; \tilde{b}_n\big(\mathbf{1}\mathbf{1}^\top - I\big)} \tag{INT–2}$$

$$\lambda_1 = \tilde{a}_n + 2\tilde{b}_n, \qquad \lambda_{2,3} = \tilde{a}_n - \tilde{b}_n. \tag{INT–3}$$

4 Organisation–wide calibration

$$\mathrm{DK}_{\mathrm{org}} = \frac{\displaystyle\int_1^5 \Delta(\omega)\,o(\omega)\,d\omega}{\displaystyle\int_1^5 o(\omega)\,d\omega} \tag{DK–2}$$

Set κ, λ as linear (or clipped) functions of $\mathrm{DK}_{\mathrm{org}}$ to tilt *all* Jacobians simultaneously.

5 Practical workflow

#	**Action**	**Feeds into**
1	Measure $\Phi(\omega),\ \Psi(\omega)\ \Rightarrow\ \Delta(\omega)$	Bias inputs
2	Derive $a_n,\ b_n$ via (J–1)	Jacobian skeleton
3	Apply (INT–1) to get $\tilde{a}_n, \tilde{b}_n$	Integrated Jacobian
4	Analyse eigen–values & simulate $\dot{x} = \tilde{J}_n x$	Stability map
5	Reduce $\lvert\Delta\rvert$ through training/hiring	Feedback loop

6 Tuning hints

- Start with $\kappa\approx 1,\ \lambda\approx 0$ to affect only cross–talk.
- Clip to $\lvert\kappa\Delta\rvert \le 1$ to keep $\tilde{b}_n \ge 0$.
- Monte–Carlo κ, λ, Δ distributions to stress–test before rolling out new LLM workflows.

How Dunning–Kruger Bias Shrinks Closed–Form Possibilities

#	Bounded analytic aspect	Why DK reduces the admissible set
1	*Spectral locality*	Bias-adjusted scale $\tilde{b}_n = b_n\,(1+\kappa\Delta)$ clips b to $b_n(1-\kappa) \le \tilde{b}_n \le b_n(1+\kappa)$. Eigen-pairs that relied on the full line segment of b no longer exist.
2	*Lyapunov candidates*	$\tilde{J}_n$ now depends on state/population via $\Delta(\omega, t)$. A single quadratic $P \succ 0$ must satisfy $P\tilde{J}_n + \tilde{J}_n^\top P \prec 0$ *for all* Δ, shrinking the feasible P to (sometimes) an empty set and forcing SOS/LMI search.
3	*Exact matrix exponentials*	Time-varying Δ makes $x(t) = \mathcal{T}\exp\big(\int \tilde{J}(t)\,dt\big)x(0)$ path-ordered; simple $\exp(\lambda t)$ forms survive only when Δ is constant or $\lvert\kappa\Delta\rvert \ll 1$ (Magnus expansion).
4	*Normal-form reducibility*	The unbiased Jacobian is "rank-one perturbed diagonal" and fully reducible. Multiplicative DK perturbations break the commutator structure, so Poincaré–Dulac or Jordan tricks require higher-order terms; closed forms disappear outside small-bias limits.

Take-away. Dunning–Kruger does not render the system unsolvable, but it *carves away* many friendly corners of the solution landscape. Analysts should expect to replace global closed forms with (i) piecewise-analytic patches where Δ is quasi-static, or (ii) perturbative/numerical treatments that respect the cognitive-bias envelope $\lvert\kappa\Delta\rvert \le 1$.