Effective Theory of Deep Learning

Beyond the Infinite-Width Limit

Dan Roberts[a] and Sho Yaida[b]

[a]MIT, IAIFI, & Salesforce, [b]Facebook AI Research

Deep Learning Theory Summer School at Princeton
July 27, 2021 – August 8, 2021

Course Plan

~~Lecture 1~~ ~~**Initialization, Linear Models**~~
- ~~§0 + §7.1 + §10.4~~

Lecture 2 **Quadratic Models & Nearly-Kernel Methods**
- §11.4 (+ §7.2) + §∞.2.2

Lecture 3 **The Principle of Sparsity (Recursing)**
- §4, §8, §11.2, §∞.3

Lecture 4 **The Principle of Criticality**
- §5, §9, §11.3, §∞.1, +§10.3

Lecture 5 **The End of Training & More**
- §∞.2.3 + Maybe { §A, §B, . . . }

Linear Models and Kernel Methods

Two forms of a solution for a **linear model**:

- *parameter space – linear regression*

$$z_i(x_{\dot\beta};\theta^\star) = \sum_{j=0}^{n_f} W^\star_{ij}\,\phi_j(x_{\dot\beta})$$

- *sample space – kernel methods*

$$z_i(x_{\dot\beta};\theta^\star) = \sum_{\tilde\alpha_1,\tilde\alpha_2\in\mathcal{A}} k_{\dot\beta\tilde\alpha_1}\,\widetilde{k}^{\tilde\alpha_1\tilde\alpha_2}\,y_{i;\tilde\alpha_2}\,.$$

Linear Models and Kernel Methods

Two forms of a solution for a **linear model**:

- *parameter space – linear regression*

$$z_i(x_{\dot\beta};\theta^\star) = \sum_{j=0}^{n_f} W^\star_{ij}\phi_j(x_{\dot\beta})$$

- *sample space – kernel methods*

$$z_i(x_{\dot\beta};\theta^\star) = \sum_{\tilde\alpha_1,\tilde\alpha_2\in\mathcal{A}} k_{\dot\beta\tilde\alpha_1}\widetilde{k}^{\tilde\alpha_1\tilde\alpha_2}y_{i;\tilde\alpha_2}\,.$$

Features of this model, expressed as $\phi_j(x)$ or $k_{\delta_1\delta_2}$, are *fixed*.
The goal of this lecture find a model where they evolve: a **minimal model of representation learning**.

Nonlinear Models

To go beyond the linear paradigm, let's slightly *deform* it to get a **nonlinear model**, specifically a **quadratic model**:

$$z_{i;\delta}(\theta) = \sum_{j=0}^{n_f} W_{ij}\phi_j(x_\delta) + \frac{\epsilon}{2}\sum_{j_1,j_2=0}^{n_f} W_{ij_1}W_{ij_2}\psi_{j_1j_2}(x_\delta)$$

Nonlinear Models

To go beyond the linear paradigm, let's slightly *deform* it to get a **nonlinear model**, specifically a **quadratic model**:

$$z_{i;\delta}(\theta) = \sum_{j=0}^{n_f} W_{ij}\phi_j(x_\delta) + \frac{\epsilon}{2}\sum_{j_1,j_2=0}^{n_f} W_{ij_1}W_{ij_2}\psi_{j_1j_2}(x_\delta)$$

- It's nonlinear because it's quadratic in the weights: $W_{ij_1}W_{ij_2}$.

Nonlinear Models

To go beyond the linear paradigm, let's slightly *deform* it to get a **nonlinear model**, specifically a **quadratic model**:

$$z_{i;\delta}(\theta) = \sum_{j=0}^{n_f} W_{ij}\phi_j(x_\delta) + \frac{\epsilon}{2}\sum_{j_1,j_2=0}^{n_f} W_{ij_1}W_{ij_2}\psi_{j_1j_2}(x_\delta)$$

- It's nonlinear because it's quadratic in the weights: $W_{ij_1}W_{ij_2}$.
- $\epsilon \ll 1$ is small parameter that controls the size of the deformation.

Nonlinear Models

To go beyond the linear paradigm, let's slightly *deform* it to get a **nonlinear model**, specifically a **quadratic model**:

$$z_{i;\delta}(\theta) = \sum_{j=0}^{n_f} W_{ij}\phi_j(x_\delta) + \frac{\epsilon}{2}\sum_{j_1,j_2=0}^{n_f} W_{ij_1}W_{ij_2}\psi_{j_1j_2}(x_\delta)$$

- It's nonlinear because it's quadratic in the weights: $W_{ij_1}W_{ij_2}$.
- $\epsilon \ll 1$ is small parameter that controls the size of the deformation.
- We've introduced $(n_f+1)(n_f+2)/2$ **meta feature functions**, $\psi_{j_1j_2}(x)$, with *two* feature indices.

Quadratic Models

To familiarize ourselves with this model, let's make a small change in the model parameters $W_{ij} \to W_{ij} + dW_{ij}$:

$$z_i(x_\delta; \theta + d\theta) = z_i(x_\delta; \theta) + \sum_{j=0}^{n_f} dW_{ij} \left[\phi_j(x_\delta) + \epsilon \sum_{j_1=0}^{n_f} W_{ij_1} \psi_{j_1 j}(x_\delta) \right] + \frac{\epsilon}{2} \sum_{j_1, j_2=0}^{n_f} dW_{ij_1} dW_{ij_2} \psi_{j_1 j_2}(x_\delta).$$

Quadratic Models

To familiarize ourselves with this model, let's make a small change in the model parameters $W_{ij} \to W_{ij} + dW_{ij}$:

$$z_i(x_\delta;\theta+d\theta) = z_i(x_\delta;\theta) + \sum_{j=0}^{n_f} dW_{ij} \left[\phi_j(x_\delta) + \epsilon \sum_{j_1=0}^{n_f} W_{ij_1}\psi_{j_1 j}(x_\delta)\right] + \frac{\epsilon}{2}\sum_{j_1,j_2=0}^{n_f} dW_{ij_1} dW_{ij_2}\psi_{j_1 j_2}(x_\delta).$$

Let us make a shorthand for the quantity in the square bracket,

$$\phi^{\mathsf{E}}_{ij}(x_\delta;\theta) \equiv \frac{dz_i(x_\delta;\theta)}{dW_{ij}} = \phi_j(x_\delta) + \epsilon \sum_{k=0}^{n_f} W_{ik}\psi_{kj}(x_\delta)\,,$$

which is an **effective feature function**.

Effective Feature Functions

The utility of this is as follows:

Effective Feature Functions

The utility of this is as follows:

- The *linear response* of $z_i(x_\delta; \theta)$ behaves *effectively* as if it has a parameter-dependent feature function, $\phi^{\mathsf{E}}_{ij}(x_\delta; \theta)$.

Effective Feature Functions

The utility of this is as follows:

- The *linear response* of $z_i(x_\delta;\theta)$ behaves *effectively* as if it has a parameter-dependent feature function, $\phi^{\mathsf{E}}_{ij}(x_\delta;\theta)$.
- The change in the $\phi^{\mathsf{E}}_{ij}(x_\delta;\theta)$ given $W_{ik} \to W_{ik} + dW_{ik}$ is

$$\phi^{\mathsf{E}}_{ij}(x_\delta;\theta+d\theta) = \phi^{\mathsf{E}}_{ij}(x_\delta;\theta) + \epsilon \sum_{k=0}^{n_f} dW_{ik}\,\psi_{kj}(x_\delta)\,,$$

Effective Feature Functions

The utility of this is as follows:

- The *linear response* of $z_i(x_\delta;\theta)$ behaves *effectively* as if it has a parameter-dependent feature function, $\phi^{\mathsf{E}}_{ij}(x_\delta;\theta)$.
- The change in the $\phi^{\mathsf{E}}_{ij}(x_\delta;\theta)$ given $W_{ik} \to W_{ik} + dW_{ik}$ is

$$\phi^{\mathsf{E}}_{ij}(x_\delta;\theta+d\theta) = \phi^{\mathsf{E}}_{ij}(x_\delta;\theta) + \epsilon \sum_{k=0}^{n_f} dW_{ik}\,\psi_{kj}(x_\delta)\,,$$

- For comparison, for the linear model we had:

$$z_i(x_\delta;\theta+d\theta) = z_i(x_\delta;\theta) + \sum_{j=0}^{n_f} dW_{ij}\,\phi_j(x_\delta)$$

Effective Feature Functions

The utility of this is as follows:

- The *linear response* of $z_i(x_\delta;\theta)$ behaves *effectively* as if it has a parameter-dependent feature function, $\phi^{\mathsf{E}}_{ij}(x_\delta;\theta)$.
- The change in the $\phi^{\mathsf{E}}_{ij}(x_\delta;\theta)$ given $W_{ik} \to W_{ik} + dW_{ik}$ is

$$\phi^{\mathsf{E}}_{ij}(x_\delta;\theta+d\theta) = \phi^{\mathsf{E}}_{ij}(x_\delta;\theta) + \epsilon \sum_{k=0}^{n_f} dW_{ik}\, \psi_{kj}(x_\delta)\,,$$

- For comparison, for the linear model we had:

$$z_i(x_\delta;\theta+d\theta) = z_i(x_\delta;\theta) + \sum_{j=0}^{n_f} dW_{ij}\, \phi_j(x_\delta)$$

Thus quadratic model has a *hierarchical structure*, where the features evolve as if they are described by a linear model and the model's output evolves in a more complicated nonlinear way.

Quadratic Regression

Supervised learning a quadratic model doesn't have a particular name, but if it did, we'd all probably agree that its name should be **quadratic regression**:

$$\mathcal{L}_{\mathcal{A}}(\theta) = \frac{1}{2} \sum_{\tilde{\alpha} \in \mathcal{A}} \sum_{i=1}^{n_{\text{out}}} \left[y_{i;\tilde{\alpha}} - \sum_{j=0}^{n_f} W_{ij}\, \phi_j(x_{\tilde{\alpha}}) - \frac{\epsilon}{2} \sum_{j_1, j_2 = 0}^{n_f} W_{ij_1} W_{ij_2} \psi_{j_1 j_2}(x_{\tilde{\alpha}}) \right]^2 .$$

Quadratic Regression

Supervised learning a quadratic model doesn't have a particular name, but if it did, we'd all probably agree that its name should be **quadratic regression**:

$$\mathcal{L}_{\mathcal{A}}(\theta) = \frac{1}{2} \sum_{\tilde{\alpha} \in \mathcal{A}} \sum_{i=1}^{n_{\text{out}}} \left[y_{i;\tilde{\alpha}} - \sum_{j=0}^{n_f} W_{ij}\, \phi_j(x_{\tilde{\alpha}}) - \frac{\epsilon}{2} \sum_{j_1, j_2 = 0}^{n_f} W_{ij_1} W_{ij_2} \psi_{j_1 j_2}(x_{\tilde{\alpha}}) \right]^2 .$$

The loss is now *quartic* in the parameters, and in general

$$0 = \left. \frac{d\mathcal{L}_{\mathcal{A}}}{dW_{ij}} \right|_{W = W^\star} ,$$

doesn't give analytical solutions or a tractable practical method.

Quadratic Regression

Supervised learning a quadratic model doesn't have a particular name, but if it did, we'd all probably agree that its name should be **quadratic regression**:

$$\mathcal{L}_{\mathcal{A}}(\theta) = \frac{1}{2} \sum_{\tilde{\alpha} \in \mathcal{A}} \sum_{i=1}^{n_{\text{out}}} \left[y_{i;\tilde{\alpha}} - \sum_{j=0}^{n_f} W_{ij}\, \phi_j(x_{\tilde{\alpha}}) - \frac{\epsilon}{2} \sum_{j_1, j_2 = 0}^{n_f} W_{ij_1} W_{ij_2} \psi_{j_1 j_2}(x_{\tilde{\alpha}}) \right]^2 .$$

The loss is now *quartic* in the parameters, but we can optimize with *gradient descent*:

$$W_{ij}(t+1) = W_{ij}(t) - \eta \frac{d\mathcal{L}_{\mathcal{A}}}{dW_{ij}}\bigg|_{W_{ij} = W_{ij}(t)} .$$

This will find a minimum in practice.

Aside: Gradient Descent

Gradient descent (GD) can be used to minimize the training loss:

$$\theta_\mu(t+1) = \theta_\mu(t) - \eta \frac{d\mathcal{L}_\mathcal{A}}{d\theta_\mu}\bigg|_{\theta_\mu=\theta_\mu(t)} .$$

Aside: Gradient Descent

Gradient descent (GD) can be used to minimize the training loss:

$$\theta_\mu(t+1) = \theta_\mu(t) - \eta \frac{d\mathcal{L}_\mathcal{A}}{d\theta_\mu}\bigg|_{\theta_\mu=\theta_\mu(t)} .$$

- GD is an *iterative* learning algorithm, and here t keeps track of the number of steps in the iterative training process.

Aside: Gradient Descent

Gradient descent (GD) can be used to minimize the training loss:

$$\theta_\mu(t+1) = \theta_\mu(t) - \eta \left. \frac{d\mathcal{L}_\mathcal{A}}{d\theta_\mu} \right|_{\theta_\mu = \theta_\mu(t)} .$$

- GD is an *iterative* learning algorithm, and here t keeps track of the number of steps in the iterative training process.
- $\eta > 0$ is a **training hyperparameter** called the **learning rate**, which controls the size of the step taken in *parameter space*.

Aside: Gradient Descent

Gradient descent (GD) can be used to minimize the training loss:

$$\theta_\mu(t+1) = \theta_\mu(t) - \eta \frac{d\mathcal{L}_\mathcal{A}}{d\theta_\mu}\bigg|_{\theta_\mu=\theta_\mu(t)} .$$

- GD is an *iterative* learning algorithm, and here t keeps track of the number of steps in the iterative training process.
- $\eta > 0$ is a **training hyperparameter** called the **learning rate**, which controls the size of the step taken in *parameter space*.
- The computational cost of gradient descent scales linearly with the size of the dataset $\mathcal{A}$, as one just needs to compute the gradient for each sample and then add them up.

Aside: Gradient Descent

For sufficiently small η, the GD updates are guaranteed to decrease the training loss $\mathcal{L}_{\mathcal{A}}$:

$$\Delta\mathcal{L}_{\mathcal{A}} \equiv \mathcal{L}_{\mathcal{A}}\Big(\theta(t+1)\Big) - \mathcal{L}_{\mathcal{A}}\Big(\theta(t)\Big)$$

Aside: Gradient Descent

For sufficiently small η, the GD updates are guaranteed to decrease the training loss $\mathcal{L}_{\mathcal{A}}$:

$$\begin{aligned}\Delta\mathcal{L}_{\mathcal{A}} &\equiv \mathcal{L}_{\mathcal{A}}\Big(\theta(t+1)\Big) - \mathcal{L}_{\mathcal{A}}\Big(\theta(t)\Big) \\ &= \mathcal{L}_{\mathcal{A}}\left(\theta_{\mu}(t) - \eta\frac{d\mathcal{L}_{\mathcal{A}}}{d\theta_{\mu}(t)}\right) - \mathcal{L}_{\mathcal{A}}\Big(\theta(t)\Big)\end{aligned}$$

Aside: Gradient Descent

For sufficiently small η, the GD updates are guaranteed to decrease the training loss $\mathcal{L}_{\mathcal{A}}$:

$$\begin{aligned}
\Delta\mathcal{L}_{\mathcal{A}} &\equiv \mathcal{L}_{\mathcal{A}}\Big(\theta(t+1)\Big) - \mathcal{L}_{\mathcal{A}}\Big(\theta(t)\Big) \\
&= \mathcal{L}_{\mathcal{A}}\left(\theta_\mu(t) - \eta\frac{d\mathcal{L}_{\mathcal{A}}}{d\theta_\mu(t)}\right) - \mathcal{L}_{\mathcal{A}}\Big(\theta(t)\Big) \\
&= -\eta\sum_\mu \left(\frac{d\mathcal{L}_{\mathcal{A}}}{d\theta_\mu}\right)^2\Bigg|_{\theta=\theta(t)} + O(\eta^2)\,.
\end{aligned}$$

Aside: Gradient Descent

For sufficiently small η, the GD updates are guaranteed to decrease the training loss $\mathcal{L}_{\mathcal{A}}$:

$$\begin{aligned}\Delta\mathcal{L}_{\mathcal{A}} &\equiv \mathcal{L}_{\mathcal{A}}\Big(\theta(t+1)\Big) - \mathcal{L}_{\mathcal{A}}\Big(\theta(t)\Big) \\ &= \mathcal{L}_{\mathcal{A}}\left(\theta_\mu(t) - \eta\frac{d\mathcal{L}_{\mathcal{A}}}{d\theta_\mu(t)}\right) - \mathcal{L}_{\mathcal{A}}\Big(\theta(t)\Big) \\ &= -\eta\sum_\mu \left(\frac{d\mathcal{L}_{\mathcal{A}}}{d\theta_\mu}\right)^2\Bigg|_{\theta=\theta(t)} + O(\eta^2)\,.\end{aligned}$$

In practice, small variants of gradient descent are responsible for almost all training and optimization in deep learning.

Aside of the Aside: Tensorial Gradient Descent

In one such variant, we modify the update as

$$\theta_\mu(t+1) = \theta_\mu(t) - \eta \sum_\nu \lambda_{\mu\nu} \frac{d\mathcal{L}_\mathcal{A}}{d\theta_\nu}\bigg|_{\theta=\theta(t)},$$

to get a more general family of learning algorithms.

Aside of the Aside: Tensorial Gradient Descent

In one such variant, we modify the update as

$$\theta_\mu(t+1) = \theta_\mu(t) - \eta \sum_\nu \lambda_{\mu\nu} \frac{d\mathcal{L}_\mathcal{A}}{d\theta_\nu}\bigg|_{\theta=\theta(t)},$$

to get a more general family of learning algorithms.

- Here $\lambda_{\mu\nu}$ is a **learning-rate tensor** on parameter space; *vanilla* GD is a special case with $\lambda_{\mu\nu} = \delta_{\mu\nu}$.

Aside of the Aside: Tensorial Gradient Descent

In one such variant, we modify the update as

$$\theta_\mu(t+1) = \theta_\mu(t) - \eta \sum_\nu \lambda_{\mu\nu} \frac{d\mathcal{L}_\mathcal{A}}{d\theta_\nu}\bigg|_{\theta=\theta(t)} ,$$

to get a more general family of learning algorithms.

- Here $\lambda_{\mu\nu}$ is a **learning-rate tensor** on parameter space; *vanilla* GD is a special case with $\lambda_{\mu\nu} = \delta_{\mu\nu}$.
- Repeating our analysis with this generalized update, we find

$$\Delta\mathcal{L}_\mathcal{A} = -\eta \sum_{\mu,\nu} \lambda_{\mu\nu} \frac{d\mathcal{L}_\mathcal{A}}{d\theta_\mu} \frac{d\mathcal{L}_\mathcal{A}}{d\theta_\nu} + O(\eta^2) ,$$

  meaning $\Delta\mathcal{L}_\mathcal{A}$ will decrease for $\eta \ll 1$, so long as the learning-rate tensor $\lambda_{\mu\nu}$ is a positive semidefinite matrix.

The Theoretical Minimum

Let's start by seeing how gradient descent solves the *linear model*:

$$\mathcal{L}_{\mathcal{A}}(W) = \frac{1}{2} \sum_{\tilde{\alpha} \in \mathcal{A}} \sum_{i=1}^{n_{\text{out}}} \left[y_{i;\tilde{\alpha}} - \sum_{j=0}^{n_f} W_{ij}\, \phi_j(x_{\tilde{\alpha}}) \right]^2 ,$$

The Theoretical Minimum

Let's start by seeing how gradient descent solves the *linear model*:

$$\mathcal{L}_{\mathcal{A}}(W) = \frac{1}{2} \sum_{\tilde{\alpha} \in \mathcal{A}} \sum_{i=1}^{n_{\text{out}}} \left[y_{i;\tilde{\alpha}} - \sum_{j=0}^{n_f} W_{ij}\, \phi_j(x_{\tilde{\alpha}}) \right]^2 ,$$

$$\frac{\partial \mathcal{L}_{\mathcal{A}}(W)}{\partial W_{ab}} = - \sum_{\tilde{\alpha}, i, j} \delta_{ia} \delta_{jb}\, \phi_j(x_{\tilde{\alpha}}) \left[y_{i;\tilde{\alpha}} - \sum_{j=0}^{n_f} W_{ij}\, \phi_j(x_{\tilde{\alpha}}) \right]$$

The Theoretical Minimum

Let's start by seeing how gradient descent solves the *linear model*:

$$\mathcal{L}_{\mathcal{A}}(W) = \frac{1}{2} \sum_{\tilde{\alpha} \in \mathcal{A}} \sum_{i=1}^{n_{\text{out}}} \left[y_{i;\tilde{\alpha}} - \sum_{j=0}^{n_f} W_{ij}\, \phi_j(x_{\tilde{\alpha}}) \right]^2 ,$$

$$\begin{aligned} \frac{\partial \mathcal{L}_{\mathcal{A}}(W)}{\partial W_{ab}} &= - \sum_{\tilde{\alpha}, i, j} \delta_{ia} \delta_{jb}\, \phi_j(x_{\tilde{\alpha}}) \left[y_{i;\tilde{\alpha}} - \sum_{j=0}^{n_f} W_{ij}\, \phi_j(x_{\tilde{\alpha}}) \right] \\ &= \sum_{\tilde{\alpha}} \phi_b(x_{\tilde{\alpha}}) \left(z_{a;\tilde{\alpha}} - y_{a;\tilde{\alpha}} \right) \end{aligned}$$

The Theoretical Minimum

Let's start by seeing how gradient descent solves the *linear model*:

$$\mathcal{L}_{\mathcal{A}}(W) = \frac{1}{2} \sum_{\tilde{\alpha} \in \mathcal{A}} \sum_{i=1}^{n_{\text{out}}} \left[y_{i;\tilde{\alpha}} - \sum_{j=0}^{n_f} W_{ij}\, \phi_j(x_{\tilde{\alpha}}) \right]^2 ,$$

$$\begin{aligned} \frac{\partial \mathcal{L}_{\mathcal{A}}(W)}{\partial W_{ab}} &= - \sum_{\tilde{\alpha}, i, j} \delta_{ia} \delta_{jb}\, \phi_j(x_{\tilde{\alpha}}) \left[y_{i;\tilde{\alpha}} - \sum_{j=0}^{n_f} W_{ij}\, \phi_j(x_{\tilde{\alpha}}) \right] \\ &= \sum_{\tilde{\alpha}} \phi_b(x_{\tilde{\alpha}}) \left(z_{a;\tilde{\alpha}} - y_{a;\tilde{\alpha}} \right) \\ &= \sum_{\tilde{\alpha}} \phi_b(x_{\tilde{\alpha}})\, \epsilon_{a;\tilde{\alpha}} \end{aligned}$$

The Theoretical Minimum

Let's start by seeing how gradient descent solves the *linear model*:

$$\mathcal{L}_{\mathcal{A}}(W) = \frac{1}{2} \sum_{\tilde{\alpha} \in \mathcal{A}} \sum_{i=1}^{n_{\text{out}}} \left[y_{i;\tilde{\alpha}} - \sum_{j=0}^{n_f} W_{ij}\, \phi_j(x_{\tilde{\alpha}}) \right]^2 ,$$

$$\begin{aligned} \frac{\partial \mathcal{L}_{\mathcal{A}}(W)}{\partial W_{ab}} &= - \sum_{\tilde{\alpha}, i, j} \delta_{ia} \delta_{jb}\, \phi_j(x_{\tilde{\alpha}}) \left[y_{i;\tilde{\alpha}} - \sum_{j=0}^{n_f} W_{ij}\, \phi_j(x_{\tilde{\alpha}}) \right] \\ &= \sum_{\tilde{\alpha}} \phi_b(x_{\tilde{\alpha}}) \left(z_{a;\tilde{\alpha}} - y_{a;\tilde{\alpha}} \right) \\ &= \sum_{\tilde{\alpha}} \phi_b(x_{\tilde{\alpha}})\, \epsilon_{a;\tilde{\alpha}} \end{aligned}$$

In the last line, we defined the **residual training error**:

$$\epsilon_{i;\tilde{\alpha}} \equiv z_{i;\tilde{\alpha}} - y_{i;\tilde{\alpha}}$$

The Theoretical Minimum

The weights will update as

$$
\begin{aligned}
W_{ij}(t+1) &= W_{ij}(t) - \eta \left. \frac{d\mathcal{L}_{\mathcal{A}}}{dW_{ij}} \right|_{W_{ij}=W_{ij}(t)} \\
&= W_{ij}(t) - \eta \sum_{\tilde{\alpha}} \phi_j(x_{\tilde{\alpha}})\, \epsilon_{i;\tilde{\alpha}}(t).
\end{aligned}
$$

The Theoretical Minimum

The weights will update as

$$W_{ij}(t+1) = W_{ij}(t) - \eta \left. \frac{d\mathcal{L}_{\mathcal{A}}}{dW_{ij}} \right|_{W_{ij}=W_{ij}(t)}$$
$$= W_{ij}(t) - \eta \sum_{\tilde{\alpha}} \phi_j(x_{\tilde{\alpha}})\, \epsilon_{i;\tilde{\alpha}}(t).$$

For the theoretical analysis, it's more convenient to understand how the output of the model updates:

$$z_{i;\delta}(t+1) = z_{i;\delta}(t) + \sum_{a,b} \frac{\partial z_{i;\delta}(t)}{\partial W_{ab}} \left[W_{ab}(t+1) - W_{ab}(t) \right] + \ldots$$

The Theoretical Minimum

The weights will update as

$$W_{ij}(t+1) = W_{ij}(t) - \eta \left. \frac{d\mathcal{L}_{\mathcal{A}}}{dW_{ij}} \right|_{W_{ij}=W_{ij}(t)}$$
$$= W_{ij}(t) - \eta \sum_{\tilde{\alpha}} \phi_j(x_{\tilde{\alpha}})\, \epsilon_{i;\tilde{\alpha}}(t).$$

For the theoretical analysis, it's more convenient to understand how the output of the model updates:

$$z_{i;\delta}(t+1) = z_{i;\delta}(t) + \sum_{a,b} \frac{\partial z_{i;\delta}(t)}{\partial W_{ab}} \Big[W_{ab}(t+1) - W_{ab}(t) \Big]$$

The Theoretical Minimum

The weights will update as

$$W_{ij}(t+1) = W_{ij}(t) - \eta \left. \frac{d\mathcal{L}_{\mathcal{A}}}{dW_{ij}} \right|_{W_{ij}=W_{ij}(t)}$$
$$= W_{ij}(t) - \eta \sum_{\tilde{\alpha}} \phi_j(x_{\tilde{\alpha}})\, \epsilon_{i;\tilde{\alpha}}(t).$$

For the theoretical analysis, it's more convenient to understand how the output of the model updates:

$$z_{i;\delta}(t+1) = z_{i;\delta}(t) + \sum_{a,b} \frac{\partial z_{i;\delta}(t)}{\partial W_{ab}} \left[-\eta \sum_{\tilde{\alpha}} \phi_b(x_{\tilde{\alpha}})\, \epsilon_{a;\tilde{\alpha}}(t) \right]$$

The Theoretical Minimum

The weights will update as

$$W_{ij}(t+1) = W_{ij}(t) - \eta \frac{d\mathcal{L}_{\mathcal{A}}}{dW_{ij}}\bigg|_{W_{ij}=W_{ij}(t)}$$
$$= W_{ij}(t) - \eta \sum_{\tilde{\alpha}} \phi_j(x_{\tilde{\alpha}})\, \epsilon_{i;\tilde{\alpha}}(t).$$

For the theoretical analysis, it's more convenient to understand how the output of the model updates:

$$z_{i;\delta}(t+1) = z_{i;\delta}(t) + \sum_{a,b} \delta_{ia}\phi_b(x_\delta)\left[- \eta \sum_{\tilde{\alpha}} \phi_b(x_{\tilde{\alpha}})\, \epsilon_{a;\tilde{\alpha}}(t)\right]$$

The Theoretical Minimum

The weights will update as

$$W_{ij}(t+1) = W_{ij}(t) - \eta \left. \frac{d\mathcal{L}_{\mathcal{A}}}{dW_{ij}} \right|_{W_{ij}=W_{ij}(t)}$$
$$= W_{ij}(t) - \eta \sum_{\tilde{\alpha}} \phi_j(x_{\tilde{\alpha}})\, \epsilon_{i;\tilde{\alpha}}(t).$$

For the theoretical analysis, it's more convenient to understand how the output of the model updates:

$$z_{i;\delta}(t+1) = z_{i;\delta}(t) - \eta \sum_{\tilde{\alpha}} \left[\sum_{b} \phi_b(x_\delta)\phi_b(x_{\tilde{\alpha}}) \right] \epsilon_{i;\tilde{\alpha}}(t)$$

The Theoretical Minimum

The weights will update as

$$W_{ij}(t+1) = W_{ij}(t) - \eta \left. \frac{d\mathcal{L}_{\mathcal{A}}}{dW_{ij}} \right|_{W_{ij}=W_{ij}(t)}$$
$$= W_{ij}(t) - \eta \sum_{\tilde{\alpha}} \phi_j(x_{\tilde{\alpha}})\, \epsilon_{i;\tilde{\alpha}}(t).$$

For the theoretical analysis, it's more convenient to understand how the output of the model updates:

$$z_{i;\delta}(t+1) = z_{i;\delta}(t) - \eta \sum_{\tilde{\alpha}} k_{\delta\tilde{\alpha}}\, \epsilon_{i;\tilde{\alpha}}(t)$$

The Theoretical Minimum

The weights will update as

$$W_{ij}(t+1) = W_{ij}(t) - \eta \left. \frac{d\mathcal{L}_{\mathcal{A}}}{dW_{ij}} \right|_{W_{ij}=W_{ij}(t)}$$
$$= W_{ij}(t) - \eta \sum_{\tilde{\alpha}} \phi_j(x_{\tilde{\alpha}})\, \epsilon_{i;\tilde{\alpha}}(t).$$

For the theoretical analysis, it's more convenient to understand how the output of the model updates:

$$z_{i;\delta}(t+1) = z_{i;\delta}(t) - \eta \sum_{\tilde{\alpha}} k_{\delta\tilde{\alpha}}\, \epsilon_{i;\tilde{\alpha}}(t)$$

- *Fixed* $k_{\delta\tilde{\alpha}}$ generates the dynamics of the model.

The Theoretical Minimum

The weights will update as

$$W_{ij}(t+1) = W_{ij}(t) - \eta \left. \frac{d\mathcal{L}_{\mathcal{A}}}{dW_{ij}} \right|_{W_{ij}=W_{ij}(t)}$$
$$= W_{ij}(t) - \eta \sum_{\tilde{\alpha}} \phi_j(x_{\tilde{\alpha}})\, \epsilon_{i;\tilde{\alpha}}(t).$$

For the theoretical analysis, it's more convenient to understand how the output of the model updates:

$$z_{i;\delta}(t+1) = z_{i;\delta}(t) - \eta \sum_{\tilde{\alpha}} k_{\delta\tilde{\alpha}}\, \epsilon_{i;\tilde{\alpha}}(t)$$

- *Fixed* $k_{\delta\tilde{\alpha}}$ generates the dynamics of the model.
- $\epsilon_{i;\tilde{\alpha}}(t)$ sources the updates for general inputs $\delta \in \mathcal{D}$.

The Theoretical Minimum

We have to solve a linear difference equation:

$$z_{i;\delta}(t+1) = z_{i;\delta}(t) - \eta \sum_{\tilde{\alpha}} k_{\delta\tilde{\alpha}}\, \epsilon_{i;\tilde{\alpha}}(t).$$

The Theoretical Minimum

We have to solve a linear difference equation:

$$z_{i;\delta}(t+1) = z_{i;\delta}(t) - \eta \sum_{\tilde{\alpha}} k_{\delta\tilde{\alpha}}\, \epsilon_{i;\tilde{\alpha}}(t).$$

Restricting to the training set, we get a first-order homogeneous linear difference equation,

$$z_{i;\tilde{\alpha}_1}(t+1) = z_{i;\tilde{\alpha}_1}(t) - \eta \sum_{\tilde{\alpha}_2} k_{\tilde{\alpha}_1\tilde{\alpha}_2}\, \epsilon_{i;\tilde{\alpha}_2}(t),$$

The Theoretical Minimum

We have to solve a linear difference equation:

$$z_{i;\delta}(t+1) = z_{i;\delta}(t) - \eta \sum_{\tilde{\alpha}} k_{\delta\tilde{\alpha}}\, \epsilon_{i;\tilde{\alpha}}(t).$$

Restricting to the training set, we get a first-order homogeneous linear difference equation,

$$z_{i;\tilde{\alpha}_1}(t+1) = z_{i;\tilde{\alpha}_1}(t) - \eta \sum_{\tilde{\alpha}_2} k_{\tilde{\alpha}_1\tilde{\alpha}_2}\, \epsilon_{i;\tilde{\alpha}_2}(t),$$

for the residual training error:

$$\epsilon_{i;\tilde{\alpha}_1}(t+1) = \epsilon_{i;\tilde{\alpha}_1}(t) - \eta \sum_{\tilde{\alpha}_2} k_{\tilde{\alpha}_1\tilde{\alpha}_2}\, \epsilon_{i;\tilde{\alpha}_2}(t),$$

The Theoretical Minimum

We can rewrite these dynamics:

$$\epsilon_{i;\tilde{\alpha}_1}(t+1) = \epsilon_{i;\tilde{\alpha}_1}(t) - \eta \sum_{\tilde{\alpha}_2} k_{\tilde{\alpha}_1\tilde{\alpha}_2}\, \epsilon_{i;\tilde{\alpha}_2}(t)$$

The Theoretical Minimum

We can rewrite these dynamics:

$$\epsilon_{i;\tilde{\alpha}_1}(t+1) = \sum_{\tilde{\alpha}_2} \left(\delta_{\tilde{\alpha}_1\tilde{\alpha}_2} - \eta k_{\tilde{\alpha}_1\tilde{\alpha}_2}\right) \epsilon_{i;\tilde{\alpha}_2}(t)$$

The Theoretical Minimum

We can rewrite these dynamics:

$$\epsilon_{i;\tilde{\alpha}_1}(t+1) = \sum_{\tilde{\alpha}_2} \left(\delta_{\tilde{\alpha}_1\tilde{\alpha}_2} - \eta k_{\tilde{\alpha}_1\tilde{\alpha}_2}\right) \epsilon_{i;\tilde{\alpha}_2}(t)$$

This is a repeated multiplication by a constant matrix:

$$\begin{aligned} U_{\tilde{\alpha}_t\tilde{\alpha}_0}(t) &\equiv \left[(\delta - \eta k)^t\right]_{\tilde{\alpha}_t\tilde{\alpha}_0} \\ &= \sum_{\tilde{\alpha}_1,\ldots,\tilde{\alpha}_{t-1}} \left(\delta_{\tilde{\alpha}_t\tilde{\alpha}_{t-1}} - \eta k_{\tilde{\alpha}_t\tilde{\alpha}_{t-1}}\right) \cdots \left(\delta_{\tilde{\alpha}_1\tilde{\alpha}_0} - \eta k_{\tilde{\alpha}_1\tilde{\alpha}_0}\right). \end{aligned}$$

The Theoretical Minimum

We can rewrite these dynamics:

$$\epsilon_{i;\tilde{\alpha}_1}(t+1) = \sum_{\tilde{\alpha}_2} \left(\delta_{\tilde{\alpha}_1\tilde{\alpha}_2} - \eta k_{\tilde{\alpha}_1\tilde{\alpha}_2}\right) \epsilon_{i;\tilde{\alpha}_2}(t)$$

This is a repeated multiplication by a constant matrix:

$$\begin{aligned} U_{\tilde{\alpha}_t\tilde{\alpha}_0}(t) &\equiv \left[(\delta - \eta k)^t\right]_{\tilde{\alpha}_t\tilde{\alpha}_0} \\ &= \sum_{\tilde{\alpha}_1,\ldots,\tilde{\alpha}_{t-1}} \left(\delta_{\tilde{\alpha}_t\tilde{\alpha}_{t-1}} - \eta k_{\tilde{\alpha}_t\tilde{\alpha}_{t-1}}\right) \cdots \left(\delta_{\tilde{\alpha}_1\tilde{\alpha}_0} - \eta k_{\tilde{\alpha}_1\tilde{\alpha}_0}\right). \end{aligned}$$

The solution is given by

$$\epsilon_{i;\tilde{\alpha}_1}(t) = \sum_{\tilde{\alpha}_2} U_{\tilde{\alpha}_1\tilde{\alpha}_2}(t)\, \epsilon_{i;\tilde{\alpha}_2}(0)\,,$$

and $U(t) \to 0$ as $t \to \infty$ so that the error vanishes: $z_{i;\tilde{\alpha}} \to y_{i;\tilde{\alpha}}$.

Theoretical Predictions

We *still* have to solve the difference equation for the test error:

$$z_{i;\delta}(t+1) = z_{i;\delta}(t) - \eta \sum_{\tilde{\alpha}} k_{\delta\tilde{\alpha}}\, \epsilon_{i;\tilde{\alpha}}(t)$$

Theoretical Predictions

We *still* have to solve the difference equation for the test error:

$$z_{i;\delta}(t) = z_{i;\delta}(0) - \sum_{\tilde{\alpha}\in\mathcal{A}} k_{\delta\tilde{\alpha}} \left\{ \eta \sum_{s=0}^{t-1} \epsilon_{i;\tilde{\alpha}}(s) \right\}$$

Theoretical Predictions

We *still* have to solve the difference equation for the test error:

$$z_{i;\delta}(\infty) = z_{i;\delta}(0) - \sum_{\tilde{\alpha}\in\mathcal{A}} k_{\delta\tilde{\alpha}} \left\{ \eta \sum_{s=0}^{\infty} \epsilon_{i;\tilde{\alpha}}(s) \right\}$$

Theoretical Predictions

We *still* have to solve the difference equation for the test error:

$$
\begin{aligned}
z_{i;\delta}(\infty) &= z_{i;\delta}(0) - \sum_{\tilde{\alpha}\in\mathcal{A}} k_{\delta\tilde{\alpha}} \left\{ \eta \sum_{s=0}^{\infty} \epsilon_{i;\tilde{\alpha}}(s) \right\} \\
&= z_{i;\delta}(0) - \sum_{\tilde{\alpha}\in\mathcal{A}} k_{\delta\tilde{\alpha}} \left\{ \eta \sum_{s=0}^{\infty} \left[\sum_{\tilde{\alpha}_1} U_{\tilde{\alpha}\tilde{\alpha}_1}(s)\, \epsilon_{i;\tilde{\alpha}_1}(0) \right] \right\}
\end{aligned}
$$

Theoretical Predictions

We *still* have to solve the difference equation for the test error:

$$
\begin{aligned}
z_{i;\delta}(\infty) &= z_{i;\delta}(0) - \sum_{\tilde{\alpha}\in\mathcal{A}} k_{\delta\tilde{\alpha}} \left\{ \eta \sum_{s=0}^{\infty} \epsilon_{i;\tilde{\alpha}}(s) \right\} \\
&= z_{i;\delta}(0) - \sum_{\tilde{\alpha}\in\mathcal{A}} k_{\delta\tilde{\alpha}} \left\{ \eta \sum_{s=0}^{\infty} \left[\sum_{\tilde{\alpha}_1} U_{\tilde{\alpha}\tilde{\alpha}_1}(s)\, \epsilon_{i;\tilde{\alpha}_1}(0) \right] \right\} \\
&= z_{i;\delta}(0) - \sum_{\tilde{\alpha},\tilde{\alpha}_1\in\mathcal{A}} k_{\delta\tilde{\alpha}} \left\{ \eta \sum_{s=0}^{\infty} \left[(\delta - \eta k)^s \right]_{\tilde{\alpha}\tilde{\alpha}_1} \right\} \epsilon_{i;\tilde{\alpha}_1}(0)
\end{aligned}
$$

Theoretical Predictions

We *still* have to solve the difference equation for the test error:

$$
\begin{aligned}
z_{i;\delta}(\infty) &= z_{i;\delta}(0) - \sum_{\tilde{\alpha}\in\mathcal{A}} k_{\delta\tilde{\alpha}} \left\{ \eta \sum_{s=0}^{\infty} \epsilon_{i;\tilde{\alpha}}(s) \right\} \\
&= z_{i;\delta}(0) - \sum_{\tilde{\alpha}\in\mathcal{A}} k_{\delta\tilde{\alpha}} \left\{ \eta \sum_{s=0}^{\infty} \left[\sum_{\tilde{\alpha}_1} U_{\tilde{\alpha}\tilde{\alpha}_1}(s)\, \epsilon_{i;\tilde{\alpha}_1}(0) \right] \right\} \\
&= z_{i;\delta}(0) - \sum_{\tilde{\alpha},\tilde{\alpha}_1\in\mathcal{A}} k_{\delta\tilde{\alpha}} \left\{ \eta \sum_{s=0}^{\infty} \left[(\delta - \eta k)^s \right]_{\tilde{\alpha}\tilde{\alpha}_1} \right\} \epsilon_{i;\tilde{\alpha}_1}(0) \\
&= z_{i;\delta}(0) - \sum_{\tilde{\alpha},\tilde{\alpha}_1\in\mathcal{A}} k_{\delta\tilde{\alpha}} \left\{ \eta \left[\delta - \left(\delta - \eta k \right) \right]^{-1} \right\}^{\tilde{\alpha}\tilde{\alpha}_1} \epsilon_{i;\tilde{\alpha}_1}(0)
\end{aligned}
$$

Theoretical Predictions

We *still* have to solve the difference equation for the test error:

$$
\begin{aligned}
z_{i;\delta}(\infty) &= z_{i;\delta}(0) - \sum_{\tilde{\alpha}\in\mathcal{A}} k_{\delta\tilde{\alpha}} \left\{ \eta \sum_{s=0}^{\infty} \epsilon_{i;\tilde{\alpha}}(s) \right\} \\
&= z_{i;\delta}(0) - \sum_{\tilde{\alpha}\in\mathcal{A}} k_{\delta\tilde{\alpha}} \left\{ \eta \sum_{s=0}^{\infty} \left[\sum_{\tilde{\alpha}_1} U_{\tilde{\alpha}\tilde{\alpha}_1}(s)\, \epsilon_{i;\tilde{\alpha}_1}(0) \right] \right\} \\
&= z_{i;\delta}(0) - \sum_{\tilde{\alpha},\tilde{\alpha}_1\in\mathcal{A}} k_{\delta\tilde{\alpha}} \left\{ \eta \sum_{s=0}^{\infty} \left[(\delta - \eta k)^s \right]_{\tilde{\alpha}\tilde{\alpha}_1} \right\} \epsilon_{i;\tilde{\alpha}_1}(0) \\
&= z_{i;\delta}(0) - \sum_{\tilde{\alpha},\tilde{\alpha}_1\in\mathcal{A}} k_{\delta\tilde{\alpha}} \left\{ \eta \left[\delta - \left(\delta - \eta k \right) \right]^{-1} \right\}^{\tilde{\alpha}\tilde{\alpha}_1} \epsilon_{i;\tilde{\alpha}_1}(0) \\
&= z_{i;\delta}(0) - \sum_{\tilde{\alpha},\tilde{\alpha}_1\in\mathcal{A}} k_{\delta\tilde{\alpha}} \widetilde{k}^{\tilde{\alpha}\tilde{\alpha}_1} \epsilon_{i;\tilde{\alpha}_1}(0)
\end{aligned}
$$

Theoretical Predictions

Compare *gradient descent* vs. the *direct optimization* solution:

$$z_{i;\delta}(\infty) = z_{i;\delta}(0) - \sum_{\tilde{\alpha},\tilde{\alpha}_1 \in \mathcal{A}} k_{\delta\tilde{\alpha}} \widetilde{k}^{\tilde{\alpha}\tilde{\alpha}_1} \epsilon_{i;\tilde{\alpha}_1}(0)$$

$$z_i\left(x_\delta; \theta^\star\right) = \sum_{\tilde{\alpha},\tilde{\alpha}_1 \in \mathcal{A}} k_{\delta\tilde{\alpha}_1} \widetilde{k}^{\tilde{\alpha}\tilde{\alpha}_1} y_{i;\tilde{\alpha}_1} \ .$$

Theoretical Predictions

Compare *gradient descent* vs. the *direct optimization* solution:

$$z_{i;\delta}(\infty) = z_{i;\delta}(0) - \sum_{\tilde{\alpha},\tilde{\alpha}_1 \in \mathcal{A}} k_{\delta\tilde{\alpha}} \widetilde{k}^{\tilde{\alpha}\tilde{\alpha}_1} \epsilon_{i;\tilde{\alpha}_1}(0)$$

$$z_i\left(x_\delta; \theta^\star\right) = \sum_{\tilde{\alpha},\tilde{\alpha}_1 \in \mathcal{A}} k_{\delta\tilde{\alpha}_1} \widetilde{k}^{\tilde{\alpha}\tilde{\alpha}_1} y_{i;\tilde{\alpha}_1} \, .$$

- The same if $z_{i;\delta}(0) = 0$, e.g. if $W_{ij}(0) = 0$.

Theoretical Predictions

Compare *gradient descent* vs. the *direct optimization* solution:

$$z_{i;\delta}(\infty) = z_{i;\delta}(0) - \sum_{\tilde{\alpha},\tilde{\alpha}_1 \in \mathcal{A}} k_{\delta\tilde{\alpha}} \widetilde{k}^{\tilde{\alpha}\tilde{\alpha}_1} \epsilon_{i;\tilde{\alpha}_1}(0)$$

$$z_i\left(x_\delta; \theta^\star\right) = \sum_{\tilde{\alpha},\tilde{\alpha}_1 \in \mathcal{A}} k_{\delta\tilde{\alpha}_1} \widetilde{k}^{\tilde{\alpha}\tilde{\alpha}_1} y_{i;\tilde{\alpha}_1} \, .$$

- The same if $z_{i;\delta}(0) = 0$, e.g. if $W_{ij}(0) = 0$.
- Otherwise, linear models have **algorithm independence**.

Theoretical Predictions

Compare *gradient descent* vs. the *direct optimization* solution:

$$z_{i;\delta}(\infty) = z_{i;\delta}(0) - \sum_{\tilde{\alpha},\tilde{\alpha}_1 \in \mathcal{A}} k_{\delta\tilde{\alpha}} \widetilde{k}^{\tilde{\alpha}\tilde{\alpha}_1} \epsilon_{i;\tilde{\alpha}_1}(0)$$

$$z_i(x_\delta; \theta^\star) = \sum_{\tilde{\alpha},\tilde{\alpha}_1 \in \mathcal{A}} k_{\delta\tilde{\alpha}_1} \widetilde{k}^{\tilde{\alpha}\tilde{\alpha}_1} y_{i;\tilde{\alpha}_1} \, .$$

- The same if $z_{i;\delta}(0) = 0$, e.g. if $W_{ij}(0) = 0$.
- Otherwise, linear models have **algorithm independence**.
- Importantly, $k_{\delta\tilde{\alpha}_1}$ is fixed, and the $\phi_i(x)$ *do not evolve*.

Quadratic Regression Returns!

Supervised learning a quadratic model doesn't have a particular name, but if it did, we'd all probably agree that its name should be **quadratic regression**:

$$\mathcal{L}_{\mathcal{A}}(\theta) = \frac{1}{2} \sum_{\tilde{\alpha} \in \mathcal{A}} \sum_{i=1}^{n_{\text{out}}} \left[y_{i;\tilde{\alpha}} - \sum_{j=0}^{n_f} W_{ij}\, \phi_j(x_{\tilde{\alpha}}) - \frac{\epsilon}{2} \sum_{j_1, j_2 = 0}^{n_f} W_{ij_1} W_{ij_2} \psi_{j_1 j_2}(x_{\tilde{\alpha}}) \right]^2 .$$

The loss is now *quartic* in the parameters, but we can optimize with *gradient descent*:

$$W_{ij}(t+1) = W_{ij}(t) - \eta \frac{d\mathcal{L}_{\mathcal{A}}}{dW_{ij}} \bigg|_{W_{ij} = W_{ij}(t)} .$$

This will find a minimum in practice.

Quadratic Model Dynamics

The weights will update as

$$W_{ij}(t+1) = W_{ij}(t) - \eta \frac{d\mathcal{L}_{\mathcal{A}}}{dW_{ij}}\bigg|_{W_{ij}=W_{ij}(t)}$$
$$= W_{ij}(t) - \eta \sum_{\tilde{\alpha}} \phi^{\mathsf{E}}_{ij;\tilde{\alpha}}(t)\, \epsilon_{i;\tilde{\alpha}}(t).$$

While the model and effective features update as

$$z_{i;\delta}(t+1) = z_{i;\delta}(t) + \sum_{j} dW_{ij}(t)\, \phi^{\mathsf{E}}_{ij;\delta}(t) + \frac{\epsilon}{2} \sum_{j_1,j_2} dW_{ij_1}(t)\, dW_{ij_2}(t)\, \psi_{j_1 j_2}(x_\delta),$$

$$\phi^{\mathsf{E}}_{ij;\delta}(t+1) = \phi^{\mathsf{E}}_{ij;\delta}(t) + \epsilon \sum_{k=0}^{n_f} dW_{ik}(t)\, \psi_{kj}(x_\delta).$$

Model Prediction Dynamics

$$
\begin{aligned}
& z_{i;\delta}(t+1) \\
=& z_{i;\delta}(t) + \sum_{j} dW_{ij}(t)\, \phi^{\mathsf{E}}_{ij;\delta}(t) + \frac{\epsilon}{2} \sum_{j_1, j_2} dW_{ij_1}(t) dW_{ij_2}(t) \psi_{j_1 j_2}(x_\delta)
\end{aligned}
$$

Model Prediction Dynamics

$$
\begin{aligned}
& z_{i;\delta}(t+1) \\
=& z_{i;\delta}(t) + \sum_j dW_{ij}(t)\, \phi^{\mathsf{E}}_{ij;\delta}(t) + \frac{\epsilon}{2} \sum_{j_1,j_2} dW_{ij_1}(t) dW_{ij_2}(t) \psi_{j_1 j_2}(x_\delta) \\
=& z_{i;\delta}(t) + \sum_j \left[-\eta \sum_{\tilde{\alpha}} \phi^{\mathsf{E}}_{ij;\tilde{\alpha}}(t)\, \epsilon_{i;\tilde{\alpha}}(t) \right] \phi^{\mathsf{E}}_{ij;\delta}(t) \\
& + \frac{\epsilon}{2} \sum_{j_1,j_2=0}^{n_f} \left[-\eta \sum_{\tilde{\alpha}_1} \phi^{\mathsf{E}}_{ij_1;\tilde{\alpha}_1}(t)\, \epsilon_{i;\tilde{\alpha}_1}(t) \right] \left[-\eta \sum_{\tilde{\alpha}_2} \phi^{\mathsf{E}}_{ij_2;\tilde{\alpha}_2}(t)\, \epsilon_{i;\tilde{\alpha}_2}(t) \right] \psi_{j_1 j_2}(x_\delta)
\end{aligned}
$$

Model Prediction Dynamics

$$
\begin{aligned}
& z_{i;\delta}(t+1) \\
=& z_{i;\delta}(t) + \sum_j dW_{ij}(t)\,\phi^{\mathsf{E}}_{ij;\delta}(t) + \frac{\epsilon}{2}\sum_{j_1,j_2} dW_{ij_1}(t)dW_{ij_2}(t)\psi_{j_1j_2}(x_\delta) \\
=& z_{i;\delta}(t) + \sum_j \left[-\eta \sum_{\tilde{\alpha}} \phi^{\mathsf{E}}_{ij;\tilde{\alpha}}(t)\,\epsilon_{i;\tilde{\alpha}}(t)\right]\phi^{\mathsf{E}}_{ij;\delta}(t) \\
& + \frac{\epsilon}{2}\sum_{j_1,j_2=0}^{n_f}\left[-\eta\sum_{\tilde{\alpha}_1}\phi^{\mathsf{E}}_{ij_1;\tilde{\alpha}_1}(t)\,\epsilon_{i;\tilde{\alpha}_1}(t)\right]\left[-\eta\sum_{\tilde{\alpha}_2}\phi^{\mathsf{E}}_{ij_2;\tilde{\alpha}_2}(t)\,\epsilon_{i;\tilde{\alpha}_2}(t)\right]\psi_{j_1j_2}(x_\delta) \\
=& z_{i;\delta}(t) - \eta\sum_{\tilde{\alpha}}\left[\sum_j \phi^{\mathsf{E}}_{ij;\delta}(t)\phi^{\mathsf{E}}_{ij;\tilde{\alpha}}(t)\right]\epsilon_{i;\tilde{\alpha}}(t) \\
& \quad + \frac{\eta^2}{2}\sum_{\tilde{\alpha}_1,\tilde{\alpha}_2}\left[\sum_{j_1,j_2}\epsilon\,\psi_{j_1j_2}(x_\delta)\,\phi^{\mathsf{E}}_{ij_1;\tilde{\alpha}_1}(t)\,\phi^{\mathsf{E}}_{ij_2;\tilde{\alpha}_2}(t)\right]\epsilon_{i;\tilde{\alpha}_1}(t)\,\epsilon_{i;\tilde{\alpha}_2}(t)
\end{aligned}
$$

Aside: Effective Kernel

To better understand this from the dual sample-space picture, let's analogously define an **effective kernel**

$$k^{\mathsf{E}}_{ii;\delta_1\delta_2}(\theta) \equiv \sum_{j=0}^{n_f} \phi^{\mathsf{E}}_{ij}(x_{\delta_1};\theta)\,\phi^{\mathsf{E}}_{ij}(x_{\delta_2};\theta)\,,$$

which measures a parameter-dependent similarity between two inputs x_{δ_1} and x_{δ_2} using our *effective features* $\phi^{\mathsf{E}}_{ij}(x_\delta;\theta)$.

Aside 2: Meta Kernel

This last line suggests that an important object worth defining is

$$\begin{aligned}\mu_{\delta_0\delta_1\delta_2} &\equiv \sum_{j_1,j_2=0}^{n_f} \epsilon\, \psi_{j_1 j_2}(x_{\delta_0})\, \phi_{j_1}(x_{\delta_1})\, \phi_{j_2}(x_{\delta_2}) \\ &= \sum_{j_1,j_2=0}^{n_f} \epsilon\, \psi_{j_1 j_2}(x_{\delta_0})\, \phi^{\mathsf{E}}_{ij_1}(x_{\delta_1};\theta)\, \phi^{\mathsf{E}}_{ij_2}(x_{\delta_2};\theta) + O\left(\epsilon^2\right)\end{aligned}$$

which we will call the **meta kernel**.

Aside 2: Meta Kernel

This last line suggests that an important object worth defining is

$$\begin{aligned}\mu_{\delta_0\delta_1\delta_2} &\equiv \sum_{j_1,j_2=0}^{n_f} \epsilon\, \psi_{j_1 j_2}(x_{\delta_0})\, \phi_{j_1}(x_{\delta_1})\, \phi_{j_2}(x_{\delta_2}) \\ &= \sum_{j_1,j_2=0}^{n_f} \epsilon\, \psi_{j_1 j_2}(x_{\delta_0})\, \phi^{\mathsf{E}}_{ij_1}(x_{\delta_1};\theta)\, \phi^{\mathsf{E}}_{ij_2}(x_{\delta_2};\theta) + O\Big(\epsilon^2\Big)\end{aligned}$$

which we will call the **meta kernel**.

- This is a *parameter-independent* tensor given entirely in terms of the fixed $\phi_j(x)$ and $\psi_{j_1 j_2}(x)$ that define the model.

Aside 2: Meta Kernel

This last line suggests that an important object worth defining is

$$\mu_{\delta_0\delta_1\delta_2} \equiv \sum_{j_1,j_2=0}^{n_f} \epsilon\, \psi_{j_1 j_2}(x_{\delta_0})\, \phi_{j_1}(x_{\delta_1})\, \phi_{j_2}(x_{\delta_2})$$
$$= \sum_{j_1,j_2=0}^{n_f} \epsilon\, \psi_{j_1 j_2}(x_{\delta_0})\, \phi^{\mathsf{E}}_{ij_1}(x_{\delta_1};\theta)\, \phi^{\mathsf{E}}_{ij_2}(x_{\delta_2};\theta) + O\Big(\epsilon^2\Big)$$

which we will call the **meta kernel**.

- This is a *parameter-independent* tensor given entirely in terms of the fixed $\phi_j(x)$ and $\psi_{j_1 j_2}(x)$ that define the model.
- For a fixed input x_{δ_0}, $\mu_{\delta_0\delta_1\delta_2}$ computes a different feature-space inner product between the two inputs, x_{δ_1} & x_{δ_2}.

Aside 2: Meta Kernel

This last line suggests that an important object worth defining is

$$\begin{aligned}\mu_{\delta_0\delta_1\delta_2} &\equiv \sum_{j_1,j_2=0}^{n_f} \epsilon\, \psi_{j_1 j_2}(x_{\delta_0})\, \phi_{j_1}(x_{\delta_1})\, \phi_{j_2}(x_{\delta_2}) \\ &= \sum_{j_1,j_2=0}^{n_f} \epsilon\, \psi_{j_1 j_2}(x_{\delta_0})\, \phi^{\mathsf{E}}_{ij_1}(x_{\delta_1};\theta)\, \phi^{\mathsf{E}}_{ij_2}(x_{\delta_2};\theta) + O\Big(\epsilon^2\Big)\end{aligned}$$

which we will call the **meta kernel**.

- This is a *parameter-independent* tensor given entirely in terms of the fixed $\phi_j(x)$ and $\psi_{j_1 j_2}(x)$ that define the model.
- For a fixed input x_{δ_0}, $\mu_{\delta_0\delta_1\delta_2}$ computes a different feature-space inner product between the two inputs, x_{δ_1} & x_{δ_2}.
- Due to the inclusion of ϵ into the definition of $\mu_{\delta_0\delta_1\delta_2}$, we should think of it as being parametrically small too.

Model Prediction Dynamics

$$
\begin{aligned}
& z_{i;\delta}(t+1) \\
=& z_{i;\delta}(t) - \eta \sum_{\tilde{\alpha}} \left[\sum_{j} \phi^{\mathsf{E}}_{ij;\delta}(t) \phi^{\mathsf{E}}_{ij;\tilde{\alpha}}(t) \right] \epsilon_{i;\tilde{\alpha}}(t) \\
& \quad + \frac{\eta^2}{2} \sum_{\tilde{\alpha}_1, \tilde{\alpha}_2} \left[\epsilon \sum_{j_1, j_2} \phi^{\mathsf{E}}_{ij_1;\tilde{\alpha}_1}(t)\, \phi^{\mathsf{E}}_{ij_2;\tilde{\alpha}_2}(t)\, \psi_{j_1 j_2}(x_\delta) \right] \epsilon_{i;\tilde{\alpha}_1}(t)\, \epsilon_{i;\tilde{\alpha}_2}(t)
\end{aligned}
$$

Model Prediction Dynamics

$$
\begin{aligned}
& z_{i;\delta}(t+1) \\
=& z_{i;\delta}(t) - \eta \sum_{\tilde{\alpha}} \left[\sum_{j} \phi^{\mathsf{E}}_{ij;\delta}(t) \phi^{\mathsf{E}}_{ij;\tilde{\alpha}}(t) \right] \epsilon_{i;\tilde{\alpha}}(t) \\
& \quad + \frac{\eta^2}{2} \sum_{\tilde{\alpha}_1, \tilde{\alpha}_2} \left[\epsilon \sum_{j_1, j_2} \phi^{\mathsf{E}}_{ij_1;\tilde{\alpha}_1}(t)\, \phi^{\mathsf{E}}_{ij_2;\tilde{\alpha}_2}(t)\, \psi_{j_1 j_2}(x_\delta) \right] \epsilon_{i;\tilde{\alpha}_1}(t)\, \epsilon_{i;\tilde{\alpha}_2}(t) \\
=& z_{i;\delta}(t) - \eta \sum_{\tilde{\alpha}} k^{\mathsf{E}}_{ii;\delta\tilde{\alpha}}(t)\, \epsilon_{i;\tilde{\alpha}}(t) + \frac{\eta^2}{2} \sum_{\tilde{\alpha}_1, \tilde{\alpha}_2} \mu_{\delta\tilde{\alpha}_1\tilde{\alpha}_2} \epsilon_{i;\tilde{\alpha}_1}(t)\, \epsilon_{i;\tilde{\alpha}_2}(t) + O\left(\epsilon^2\right)
\end{aligned}
$$

Model Prediction Dynamics

$$
\begin{aligned}
& z_{i;\delta}(t+1) \\
=& z_{i;\delta}(t) - \eta \sum_{\tilde{\alpha}} \left[\sum_{j} \phi^{\mathsf{E}}_{ij;\delta}(t) \phi^{\mathsf{E}}_{ij;\tilde{\alpha}}(t) \right] \epsilon_{i;\tilde{\alpha}}(t) \\
& \quad + \frac{\eta^2}{2} \sum_{\tilde{\alpha}_1, \tilde{\alpha}_2} \left[\epsilon \sum_{j_1, j_2} \phi^{\mathsf{E}}_{ij_1;\tilde{\alpha}_1}(t)\, \phi^{\mathsf{E}}_{ij_2;\tilde{\alpha}_2}(t)\, \psi_{j_1 j_2}(x_\delta) \right] \epsilon_{i;\tilde{\alpha}_1}(t)\, \epsilon_{i;\tilde{\alpha}_2}(t) \\
=& z_{i;\delta}(t) - \eta \sum_{\tilde{\alpha}} k^{\mathsf{E}}_{ii;\delta\tilde{\alpha}}(t)\, \epsilon_{i;\tilde{\alpha}}(t) + \frac{\eta^2}{2} \sum_{\tilde{\alpha}_1, \tilde{\alpha}_2} \mu_{\delta\tilde{\alpha}_1\tilde{\alpha}_2} \epsilon_{i;\tilde{\alpha}_1}(t)\, \epsilon_{i;\tilde{\alpha}_2}(t) + O\Big(\epsilon^2\Big)
\end{aligned}
$$

This is a coupled nonlinear difference equation. . .

Effective Kernel Dynamics

$$\phi^{\mathsf{E}}_{ij;\delta}(t+1) = \phi^{\mathsf{E}}_{ij;\delta}(t) + \epsilon \sum_{k=0}^{n_f} dW_{ik}(t)\, \psi_{kj}(x_\delta)$$

Effective Kernel Dynamics

$$\phi^{\mathsf{E}}_{ij;\delta}(t+1) = \phi^{\mathsf{E}}_{ij;\delta}(t) + \epsilon \sum_{k=0}^{n_f} \left[-\eta \sum_{\tilde{\alpha}} \phi^{\mathsf{E}}_{ik;\tilde{\alpha}}(t)\, \epsilon_{i;\tilde{\alpha}}(t) \right] \psi_{kj}(x_\delta)$$

Effective Kernel Dynamics

$$\phi^{\mathsf{E}}_{ij;\delta}(t+1) = \phi^{\mathsf{E}}_{ij;\delta}(t) - \eta \sum_{\tilde{\alpha}} \left[\sum_{k=0}^{n_f} \epsilon \, \psi_{kj}(x_\delta) \, \phi^{\mathsf{E}}_{ik;\tilde{\alpha}}(t) \right] \epsilon_{i;\tilde{\alpha}}(t)$$

Effective Kernel Dynamics

$$\phi^{\mathsf{E}}_{ij;\delta}(t+1) = \phi^{\mathsf{E}}_{ij;\delta}(t) - \eta \sum_{\tilde{\alpha}} \left[\sum_{k=0}^{n_f} \epsilon \, \psi_{kj}(x_\delta) \, \phi^{\mathsf{E}}_{ik;\tilde{\alpha}}(t) \right] \epsilon_{i;\tilde{\alpha}}(t)$$

$$\Longrightarrow$$

Effective Kernel Dynamics

$$\phi^{\mathsf{E}}_{ij;\delta}(t+1) = \phi^{\mathsf{E}}_{ij;\delta}(t) - \eta \sum_{\tilde{\alpha}} \left[\sum_{k=0}^{n_f} \epsilon \, \psi_{kj}(x_\delta) \, \phi^{\mathsf{E}}_{ik;\tilde{\alpha}}(t) \right] \epsilon_{i;\tilde{\alpha}}(t)$$

$$\Longrightarrow$$

$$\sum_j \phi^{\mathsf{E}}_{ij;\delta_1}(t+1) \, \phi^{\mathsf{E}}_{ij;\delta_2}(t+1)$$

$$= \sum_j \phi^{\mathsf{E}}_{ij;\delta_1}(t) \, \phi^{\mathsf{E}}_{ij;\delta_2}(t) - \eta \sum_{\tilde{\alpha}} \left[\sum_{j,k} \epsilon \, \psi_{kj}(x_{\delta_1}) \, \phi^{\mathsf{E}}_{ij;\delta_2}(t) \, \phi^{\mathsf{E}}_{ik;\tilde{\alpha}}(t) \right] \epsilon_{i;\tilde{\alpha}}(t)$$

$$- \eta \sum_{\tilde{\alpha}} \left[\sum_{j,k} \epsilon \, \psi_{kj}(x_{\delta_2}) \, \phi^{\mathsf{E}}_{ij;\delta_1}(t) \, \phi^{\mathsf{E}}_{ik;\tilde{\alpha}}(t) \right] \epsilon_{i;\tilde{\alpha}}(t) + O\left(\epsilon^2\right)$$

Effective Kernel Dynamics

$$\phi^{\mathsf{E}}_{ij;\delta}(t+1) = \phi^{\mathsf{E}}_{ij;\delta}(t) - \eta \sum_{\tilde{\alpha}} \left[\sum_{k=0}^{n_f} \epsilon\, \psi_{kj}(x_\delta)\, \phi^{\mathsf{E}}_{ik;\tilde{\alpha}}(t) \right] \epsilon_{i;\tilde{\alpha}}(t)$$

$$\Longrightarrow$$

$$\sum_j \phi^{\mathsf{E}}_{ij;\delta_1}(t+1)\, \phi^{\mathsf{E}}_{ij;\delta_2}(t+1)$$

$$= \sum_j \phi^{\mathsf{E}}_{ij;\delta_1}(t)\, \phi^{\mathsf{E}}_{ij;\delta_2}(t) - \eta \sum_{\tilde{\alpha}} \left[\sum_{j,k} \epsilon\, \psi_{kj}(x_{\delta_1})\, \phi_j(x_{\delta_2})\, \phi_k(x_{\tilde{\alpha}}) \right] \epsilon_{i;\tilde{\alpha}}(t)$$

$$- \eta \sum_{\tilde{\alpha}} \left[\sum_{j,k} \epsilon\, \psi_{kj}(x_{\delta_2})\, \phi_j(x_{\delta_1})\, \phi_k(x_{\tilde{\alpha}}) \right] \epsilon_{i;\tilde{\alpha}}(t) + O\left(\epsilon^2\right)$$

Effective Kernel Dynamics

$$\phi^{\mathsf{E}}_{ij;\delta}(t+1) = \phi^{\mathsf{E}}_{ij;\delta}(t) - \eta \sum_{\tilde{\alpha}} \left[\sum_{k=0}^{n_f} \epsilon \, \psi_{kj}(x_\delta) \, \phi^{\mathsf{E}}_{ik;\tilde{\alpha}}(t) \right] \epsilon_{i;\tilde{\alpha}}(t)$$

$$\Longrightarrow$$

$$k^{\mathsf{E}}_{ii;\delta_1\delta_2}(t+1) = k^{\mathsf{E}}_{ii;\delta_1\delta_2}(t) - \eta \sum_{\tilde{\alpha}} \Big(\mu_{\delta_1\delta_2\tilde{\alpha}} + \mu_{\delta_2\delta_1\tilde{\alpha}} \Big) \epsilon_{i;\tilde{\alpha}}(t) + O\Big(\epsilon^2\Big)$$

Effective Kernel Dynamics

$$\phi^{\mathsf{E}}_{ij;\delta}(t+1) = \phi^{\mathsf{E}}_{ij;\delta}(t) - \eta \sum_{\tilde{\alpha}} \left[\sum_{k=0}^{n_f} \epsilon \, \psi_{kj}(x_\delta) \, \phi^{\mathsf{E}}_{ik;\tilde{\alpha}}(t) \right] \epsilon_{i;\tilde{\alpha}}(t)$$

$$\Longrightarrow$$

$$k^{\mathsf{E}}_{ii;\delta_1\delta_2}(t+1) = k^{\mathsf{E}}_{ii;\delta_1\delta_2}(t) - \eta \sum_{\tilde{\alpha}} \Big(\mu_{\delta_1\delta_2\tilde{\alpha}} + \mu_{\delta_2\delta_1\tilde{\alpha}} \Big) \epsilon_{i;\tilde{\alpha}}(t) + O\Big(\epsilon^2\Big)$$

Linear difference equation, w/ $\mu_{\delta_1\delta_2\tilde{\alpha}}$ *playing the role of* $k_{\delta\tilde{\alpha}}$ *. . .*

Quadratic Model Dynamics: Dual Sample Space

The *model predictions* will update as

$$\begin{aligned} & z_{i;\delta}(t+1) \\ =& z_{i;\delta}(t) - \eta \sum_{\tilde{\alpha}} k^{\mathsf{E}}_{ii;\delta\tilde{\alpha}}(t)\, \epsilon_{i;\tilde{\alpha}}(t) + \frac{\eta^2}{2} \sum_{\tilde{\alpha}_1,\tilde{\alpha}_2} \mu_{\delta\tilde{\alpha}_1\tilde{\alpha}_2} \epsilon_{i;\tilde{\alpha}_1}(t)\, \epsilon_{i;\tilde{\alpha}_2}(t) + O\Big(\epsilon^2\Big) \end{aligned}$$

While the *effective kernel* will update as

$$k^{\mathsf{E}}_{ii;\delta_1\delta_2}(t+1) = k^{\mathsf{E}}_{ii;\delta_1\delta_2}(t) - \eta \sum_{\tilde{\alpha}} \Big(\mu_{\delta_1\delta_2\tilde{\alpha}} + \mu_{\delta_2\delta_1\tilde{\alpha}}\Big) \epsilon_{i;\tilde{\alpha}}(t) + O\Big(\epsilon^2\Big)$$

Quadratic Model Dynamics: Dual Sample Space

The *model predictions* will update as

$$\begin{aligned} & z_{i;\delta}(t+1) \\ =& z_{i;\delta}(t) - \eta \sum_{\tilde{\alpha}} k^{\mathsf{E}}_{ii;\delta\tilde{\alpha}}(t)\, \epsilon_{i;\tilde{\alpha}}(t) + \frac{\eta^2}{2} \sum_{\tilde{\alpha}_1,\tilde{\alpha}_2} \mu_{\delta\tilde{\alpha}_1\tilde{\alpha}_2} \epsilon_{i;\tilde{\alpha}_1}(t)\, \epsilon_{i;\tilde{\alpha}_2}(t) + O\Big(\epsilon^2\Big) \end{aligned}$$

While the *effective kernel* will update as

$$k^{\mathsf{E}}_{ii;\delta_1\delta_2}(t+1) = k^{\mathsf{E}}_{ii;\delta_1\delta_2}(t) - \eta \sum_{\tilde{\alpha}} \Big(\mu_{\delta_1\delta_2\tilde{\alpha}} + \mu_{\delta_2\delta_1\tilde{\alpha}}\Big) \epsilon_{i;\tilde{\alpha}}(t) + O\Big(\epsilon^2\Big)$$

- These joint updates are coupled *difference equations*, and the first is *nonlinear* in the training error.

Quadratic Model Dynamics: Dual Sample Space

The *model predictions* will update as

$$
\begin{aligned}
& z_{i;\delta}(t+1) \\
=& z_{i;\delta}(t) - \eta \sum_{\tilde{\alpha}} k^{\mathsf{E}}_{ii;\delta\tilde{\alpha}}(t)\, \epsilon_{i;\tilde{\alpha}}(t) + \frac{\eta^2}{2} \sum_{\tilde{\alpha}_1,\tilde{\alpha}_2} \mu_{\delta\tilde{\alpha}_1\tilde{\alpha}_2} \epsilon_{i;\tilde{\alpha}_1}(t)\, \epsilon_{i;\tilde{\alpha}_2}(t) + O\Big(\epsilon^2\Big)
\end{aligned}
$$

While the *effective kernel* will update as

$$
k^{\mathsf{E}}_{ii;\delta_1\delta_2}(t+1) = k^{\mathsf{E}}_{ii;\delta_1\delta_2}(t) - \eta \sum_{\tilde{\alpha}} \Big(\mu_{\delta_1\delta_2\tilde{\alpha}} + \mu_{\delta_2\delta_1\tilde{\alpha}}\Big) \epsilon_{i;\tilde{\alpha}}(t) + O\Big(\epsilon^2\Big)
$$

- These joint updates are coupled *difference equations*, and the first is *nonlinear* in the training error.
- We are now going to solve these equations in a closed form to leading order in ϵ using **perturbation theory**.

Another Theoretical Minimum

Decompose the model prediction into *free* and *interacting* parts:

$$z_{i;\delta}(t) \equiv z^{\mathsf{F}}_{i;\delta}(t) + z^{\mathsf{I}}_{i;\delta}(t)\,.$$

Another Theoretical Minimum

Decompose the model prediction into *free* and *interacting* parts:

$$z_{i;\delta}(t) \equiv z^{\mathsf{F}}_{i;\delta}(t) + z^{\mathsf{I}}_{i;\delta}(t)\,.$$

- $z^{\mathsf{F}}_{i;\delta}(t)$ solves the linear model dynamics from before:

$$\begin{aligned} z^{\mathsf{F}}_{i;\delta}(t+1) &= z^{\mathsf{F}}_{i;\delta}(t) - \eta \sum_{\tilde{\alpha}} k_{\delta\tilde{\alpha}}\, \epsilon^{\mathsf{F}}_{i;\tilde{\alpha}}(t) \\ &= z^{\mathsf{F}}_{i;\delta}(t) - \eta \sum_{\tilde{\alpha}} k_{\delta\tilde{\alpha}} \left[z^{\mathsf{F}}_{i;\tilde{\alpha}}(t) - y_{i;\tilde{\alpha}} \right] \,. \end{aligned}$$

Another Theoretical Minimum

Decompose the model prediction into *free* and *interacting* parts:

$$z_{i;\delta}(t) \equiv z^{\mathsf{F}}_{i;\delta}(t) + z^{\mathsf{I}}_{i;\delta}(t)\,.$$

- $z^{\mathsf{F}}_{i;\delta}(t)$ solves the linear model dynamics from before:

$$\begin{aligned} z^{\mathsf{F}}_{i;\delta}(t+1) &= z^{\mathsf{F}}_{i;\delta}(t) - \eta \sum_{\tilde{\alpha}} k_{\delta\tilde{\alpha}}\, \epsilon^{\mathsf{F}}_{i;\tilde{\alpha}}(t) \\ &= z^{\mathsf{F}}_{i;\delta}(t) - \eta \sum_{\tilde{\alpha}} k_{\delta\tilde{\alpha}} \left[z^{\mathsf{F}}_{i;\tilde{\alpha}}(t) - y_{i;\tilde{\alpha}} \right] . \end{aligned}$$

- $z^{\mathsf{I}}_{i;\delta}(t) = O(\epsilon)$, since $\epsilon \to 0$ gives back the linear model.

Another Theoretical Minimum

Let's solve the effective kernel equation first:

$$
\begin{aligned}
& k^{\mathrm{E}}_{ii;\delta_1\delta_2}(t+1) \\
=& k^{\mathrm{E}}_{ii;\delta_1\delta_2}(t) - \eta \sum_{\tilde{\alpha}} \Big(\mu_{\delta_1\delta_2\tilde{\alpha}} + \mu_{\delta_2\delta_1\tilde{\alpha}} \Big) \epsilon_{i;\tilde{\alpha}}(t) + O\Big(\epsilon^2\Big)
\end{aligned}
$$

Another Theoretical Minimum

Let's solve the effective kernel equation first:

$$
\begin{aligned}
& k^{\mathsf{E}}_{ii;\delta_1\delta_2}(t+1) \\
=& k^{\mathsf{E}}_{ii;\delta_1\delta_2}(t) - \eta \sum_{\tilde{\alpha}} \Big(\mu_{\delta_1\delta_2\tilde{\alpha}} + \mu_{\delta_2\delta_1\tilde{\alpha}} \Big) \Big[z^{\mathsf{F}}_{i;\tilde{\alpha}}(t) + z^{\mathsf{I}}_{i;\tilde{\alpha}}(t) - y_{i;\tilde{\alpha}} \Big] + O\Big(\epsilon^2\Big)
\end{aligned}
$$

Another Theoretical Minimum

Let's solve the effective kernel equation first:

$$
\begin{aligned}
& k^{\mathsf{E}}_{ii;\delta_1\delta_2}(t+1) \\
=& k^{\mathsf{E}}_{ii;\delta_1\delta_2}(t) - \eta \sum_{\tilde{\alpha}} \left(\mu_{\delta_1\delta_2\tilde{\alpha}} + \mu_{\delta_2\delta_1\tilde{\alpha}} \right) \left[z^{\mathsf{F}}_{i;\tilde{\alpha}}(t) + z^{\mathsf{I}}_{i;\tilde{\alpha}}(t) - y_{i;\tilde{\alpha}} \right]
\end{aligned}
$$

Another Theoretical Minimum

Let's solve the effective kernel equation first:

$$
\begin{aligned}
& k^{\mathsf{E}}_{ii;\delta_1\delta_2}(t+1) \\
=& k^{\mathsf{E}}_{ii;\delta_1\delta_2}(t) - \eta \sum_{\tilde{\alpha}} \left(\mu_{\delta_1\delta_2\tilde{\alpha}} + \mu_{\delta_2\delta_1\tilde{\alpha}} \right) \left[z^{\mathsf{F}}_{i;\tilde{\alpha}}(t) - y_{i;\tilde{\alpha}} \right]
\end{aligned}
$$

Another Theoretical Minimum

Let's solve the effective kernel equation first:

$$
\begin{aligned}
& k^{\mathsf{E}}_{ii;\delta_1\delta_2}(t) \\
=& k^{\mathsf{E}}_{ii;\delta_1\delta_2}(0) - \sum_{\tilde{\alpha}} \left(\mu_{\delta_1\delta_2\tilde{\alpha}} + \mu_{\delta_2\delta_1\tilde{\alpha}} \right) \left\{ \eta \sum_{s=0}^{t-1} \left[z^{\mathsf{F}}_{i;\tilde{\alpha}}(s) - y_{i;\tilde{\alpha}} \right] \right\}
\end{aligned}
$$

Another Theoretical Minimum

Let's solve the effective kernel equation first:

$$
\begin{aligned}
& k^{\mathsf{E}}_{ii;\delta_1\delta_2}(t) \\
=& k^{\mathsf{E}}_{ii;\delta_1\delta_2}(0) - \sum_{\tilde{\alpha}} \left(\mu_{\delta_1\delta_2\tilde{\alpha}} + \mu_{\delta_2\delta_1\tilde{\alpha}} \right) a_{i;\tilde{\alpha}}(t)
\end{aligned}
$$

"$a_{i;\tilde{\alpha}}$" Dynamical Helper Function

The explicit representation of $a_{i;\tilde{\alpha}}(t)$ is given by

$$
\begin{aligned}
& a_{i;\tilde{\alpha}}(t) \\
\equiv & \eta \sum_{s=0}^{t-1} \left[z^{\mathrm{F}}_{i;\tilde{\alpha}}(s) - y_{i;\tilde{\alpha}} \right] \\
= & \eta \sum_{\tilde{\alpha}_1} \left\{ \sum_{s=0}^{t-1} \left[\left(\delta - \eta \widetilde{k} \right)^s \right]_{\tilde{\alpha}\tilde{\alpha}_1} \right\} \left(z_{i;\tilde{\alpha}_1} - y_{i;\tilde{\alpha}_1} \right) \\
= & \eta \sum_{\tilde{\alpha}_1, \tilde{\alpha}_2} \left\{ \left[\delta - \left(I - \eta \widetilde{k} \right) \right]^{-1} \right\}^{\tilde{\alpha}\tilde{\alpha}_2} \left[\delta - \left(\delta - \eta \widetilde{k} \right)^t \right]_{\tilde{\alpha}_2\tilde{\alpha}_1} \left(z_{i;\tilde{\alpha}_1} - y_{i;\tilde{\alpha}_1} \right) \\
= & \sum_{\tilde{\alpha}_2} \widetilde{k}^{\tilde{\alpha}\tilde{\alpha}_2} \left\{ z_{i;\tilde{\alpha}_2} - y_{i;\tilde{\alpha}_2} - \left[z^{\mathrm{F}}_{i;\tilde{\alpha}_2}(t) - y_{i;\tilde{\alpha}_2} \right] \right\} \ .
\end{aligned}
$$

Another Theoretical Minimum

Let's solve the *interacting part* of the model prediction, $z^{\mathsf{I}}_{i;\delta}(t)$:

$$z^{\mathsf{I}}_{i;\delta}(t+1) = z^{\mathsf{I}}_{i;\delta}(t) - \sum_{j,\tilde{\alpha}} \eta\, k_{\delta\tilde{\alpha}}\, z^{\mathsf{I}}_{j;\tilde{\alpha}}(t) + \eta\, \mathbb{F}_{i;\delta}(t)\,.$$

Another Theoretical Minimum

Let's solve the *interacting part* of the model prediction, $z^{\mathsf{I}}_{i;\delta}(t)$:

$$z^{\mathsf{I}}_{i;\delta}(t+1) = z^{\mathsf{I}}_{i;\delta}(t) - \sum_{j,\tilde{\alpha}} \eta\, k_{\delta\tilde{\alpha}}\, z^{\mathsf{I}}_{j;\tilde{\alpha}}(t) + \eta\, \mathbb{F}_{i;\delta}(t)\,.$$

- We used the fact that the *free part* satisfies the free dynamics.

Another Theoretical Minimum

Let's solve the *interacting part* of the model prediction, $z^{\mathsf{I}}_{i;\delta}(t)$:

$$z^{\mathsf{I}}_{i;\delta}(t+1) = z^{\mathsf{I}}_{i;\delta}(t) - \sum_{j,\tilde{\alpha}} \eta\, k_{\delta\tilde{\alpha}}\, z^{\mathsf{I}}_{j;\tilde{\alpha}}(t) + \eta\, \mathbb{F}_{i;\delta}(t)\,.$$

- We used the fact that the *free part* satisfies the free dynamics.
- We've defined a *damping force*:

$$\mathbb{F}_{i;\delta}(t) \equiv - \sum_{\tilde{\alpha}} \left[k^{\mathsf{E}}_{\delta\tilde{\alpha}}(t) - k_{\delta\tilde{\alpha}} \right] \epsilon^{\mathsf{F}}_{i;\tilde{\alpha}}(t) + \frac{\eta}{2} \sum_{\tilde{\alpha}_1,\tilde{\alpha}_2} \mu_{\delta\tilde{\alpha}_1\tilde{\alpha}_2}\, \epsilon^{\mathsf{F}}_{i;\tilde{\alpha}_1}(t)\, \epsilon^{\mathsf{F}}_{i;\tilde{\alpha}_2}(t)$$

Another Theoretical Minimum

Let's solve the *interacting part* of the model prediction, $z^{\mathsf{I}}_{i;\delta}(t)$:

$$z^{\mathsf{I}}_{i;\delta}(t+1) = z^{\mathsf{I}}_{i;\delta}(t) - \sum_{j,\tilde{\alpha}} \eta\, k_{\delta\tilde{\alpha}}\, z^{\mathsf{I}}_{j;\tilde{\alpha}}(t) + \eta\, \mathbb{F}_{i;\delta}(t)\,.$$

- We used the fact that the *free part* satisfies the free dynamics.
- We've defined a *damping force*:

$$\begin{aligned}\mathbb{F}_{i;\delta}(t) \equiv & - \sum_{\tilde{\alpha}} \left[k^{\mathsf{E}}_{\delta\tilde{\alpha}}(t) - k_{\delta\tilde{\alpha}} \right] \epsilon^{\mathsf{F}}_{i;\tilde{\alpha}}(t) \\ & + \frac{\eta}{2} \sum_{\tilde{\alpha}_1,\tilde{\alpha}_2} \mu_{\delta\tilde{\alpha}_1\tilde{\alpha}_2}\, \epsilon^{\mathsf{F}}_{i;\tilde{\alpha}_1}(t)\, \epsilon^{\mathsf{F}}_{i;\tilde{\alpha}_2}(t)\end{aligned}$$

- Importantly, we've explicit expressions for all in $\mathbb{F}_{i;\delta}(t)$.

Another Theoretical Minimum

Let's solve the *interacting part* of the model prediction, $z^{\mathsf{I}}_{i;\delta}(t)$:

$$z^{\mathsf{I}}_{i;\delta}(t+1) = z^{\mathsf{I}}_{i;\delta}(t) - \sum_{j,\tilde{\alpha}} \eta\, k_{\delta\tilde{\alpha}}\, z^{\mathsf{I}}_{j;\tilde{\alpha}}(t) + \eta\, \mathbb{F}_{i;\delta}(t)\,.$$

Restricting to $\tilde{\alpha} \in \mathcal{A}$, we get a first-order inhomogeneous linear difference equation, with the damping force ruining the homogeneity:

$$z^{\mathsf{I}}_{i;\tilde{\alpha}}(t+1) = \sum_{\tilde{\alpha}_1} (\delta_{\tilde{\alpha}\tilde{\alpha}_1} - \eta k_{\tilde{\alpha}\tilde{\alpha}_1})\, z^{\mathsf{I}}_{i;\tilde{\alpha}_1}(t) + \eta\, \mathbb{F}_{i;\tilde{\alpha}}(t)\,.$$

Another Theoretical Minimum

Let's solve the *interacting part* of the model prediction, $z^{\mathrm{I}}_{i;\delta}(t)$:

$$z^{\mathrm{I}}_{i;\delta}(t+1) = z^{\mathrm{I}}_{i;\delta}(t) - \sum_{j,\tilde{\alpha}} \eta\, k_{\delta\tilde{\alpha}}\, z^{\mathrm{I}}_{j;\tilde{\alpha}}(t) + \eta\, \mathbb{F}_{i;\delta}(t)\,.$$

Restricting to $\tilde{\alpha} \in \mathcal{A}$, we get a first-order inhomogeneous linear difference equation, with the damping force ruining the homogeneity:

$$z^{\mathrm{I}}_{i;\tilde{\alpha}}(t+1) = \sum_{\tilde{\alpha}_1} (\delta_{\tilde{\alpha}\tilde{\alpha}_1} - \eta k_{\tilde{\alpha}\tilde{\alpha}_1})\, z^{\mathrm{I}}_{i;\tilde{\alpha}_1}(t) + \eta\, \mathbb{F}_{i;\tilde{\alpha}}(t)\,.$$

We can formally solve this by convoluting $\mathbb{F}_{i;\tilde{\alpha}}(t)$ with $U(t)$:

$$z^{\mathrm{I}}_{i;\tilde{\alpha}}(t) = \eta \sum_{s=0}^{t-1} \sum_{\tilde{\alpha}_1} U_{\tilde{\alpha}\tilde{\alpha}_1}(t-1-s)\, \mathbb{F}_{i;\tilde{\alpha}_1}(s)\,.$$

Another Theoretical Minimum

Let's solve the *interacting part* of the model prediction, $z^{\mathsf{I}}_{i;\delta}(t)$:

$$z^{\mathsf{I}}_{i;\delta}(t+1) = z^{\mathsf{I}}_{i;\delta}(t) - \sum_{j,\tilde{\alpha}} \eta\, k_{\delta\tilde{\alpha}}\, z^{\mathsf{I}}_{j;\tilde{\alpha}}(t) + \eta\, \mathbb{F}_{i;\delta}(t)\,.$$

For general inputs $\delta \in \mathcal{D}$, we can iterate to find:

$$z^{\mathsf{I}}_{i;\delta}(t) = \eta \sum_{s=0}^{t-1} \left[\mathbb{F}_{i;\delta}(s) - \sum_{\tilde{\alpha}} k_{\delta\tilde{\alpha}}\, z^{\mathsf{I}}_{i;\tilde{\alpha}}(s) \right] .$$

Another Theoretical Minimum

Let's solve the *interacting part* of the model prediction, $z^{\mathsf{I}}_{i;\delta}(t)$:

$$z^{\mathsf{I}}_{i;\delta}(t+1) = z^{\mathsf{I}}_{i;\delta}(t) - \sum_{j,\tilde{\alpha}} \eta\, k_{\delta\tilde{\alpha}}\, z^{\mathsf{I}}_{j;\tilde{\alpha}}(t) + \eta\, \mathbb{F}_{i;\delta}(t)\,.$$

For general inputs $\delta \in \mathcal{D}$, we can iterate to find:

$$z^{\mathsf{I}}_{i;\delta}(t) = \eta \sum_{s=0}^{t-1} \left[\mathbb{F}_{i;\delta}(s) - \sum_{\tilde{\alpha}} k_{\delta\tilde{\alpha}}\, z^{\mathsf{I}}_{i;\tilde{\alpha}}(s) \right] .$$

After some manipulations of this sum to be done in the privacy of your own notebook, we get a slightly less formal solution:

$$z^{\mathsf{I}}_{i;\delta}(t) = \eta \sum_{s=0}^{t-1} \left[\mathbb{F}_{i;\delta}(s) - \sum_{\tilde{\alpha}_1,\tilde{\alpha}_2} k_{\delta\tilde{\alpha}_1}\, \widetilde{k}^{\tilde{\alpha}_1\tilde{\alpha}_2} \mathbb{F}_{i;\tilde{\alpha}_2}(s) \right] + \sum_{\tilde{\alpha}_1,\tilde{\alpha}_2} k_{\delta\tilde{\alpha}_1}\, \widetilde{k}^{\tilde{\alpha}_1\tilde{\alpha}_2} z^{\mathsf{I}}_{i;\tilde{\alpha}_2}(t)\,.$$

Aside: Convergence on Training Set

Ultimately, we want $z^{\mathsf{I}}_{i;\delta}(t)$ at the end of training, $t \to \infty$.

Aside: Convergence on Training Set

Ultimately, we want $z^{\mathsf{I}}_{i;\delta}(t)$ at the end of training, $t \to \infty$.

- Before we showed that for small enough η we have:

$$\lim_{t\to\infty} U(t) \propto \exp\Big(-\eta\widetilde{k}t\Big) \ .$$

Aside: Convergence on Training Set

Ultimately, we want $z^{\mathsf{I}}_{i;\delta}(t)$ at the end of training, $t \to \infty$.

- Before we showed that for small enough η we have:

$$\lim_{t\to\infty} U(t) \propto \exp\Big(-\eta\widetilde{k}t\Big) \ .$$

Then, for training inputs, $z^{\mathsf{I}}_{i;\tilde{\alpha}}(t)$, will exponentially converge

$$\lim_{t\to\infty} z^{\mathsf{I}}_{i;\tilde{\alpha}}(t) = 0 \, .$$

Aside: Convergence on Training Set

Ultimately, we want $z^{\mathsf{I}}_{i;\delta}(t)$ at the end of training, $t \to \infty$.

- Before we showed that for small enough η we have:

$$\lim_{t\to\infty} U(t) \propto \exp\Big(-\eta\widetilde{k}t\Big) \,.$$

Then, for training inputs, $z^{\mathsf{I}}_{i;\tilde{\alpha}}(t)$, will exponentially converge

$$\lim_{t\to\infty} z^{\mathsf{I}}_{i;\tilde{\alpha}}(t) = 0 \,.$$

- To see why this holds, note that we have

$$\lim_{s\to\infty} \mathbb{F}(s) \propto \exp\Big(-\eta\widetilde{k}s\Big) \,,$$

since its leading behavior is linearly proportional to $\epsilon^{\mathsf{F}}_{j;\tilde{\alpha}}(t)$.

Aside: Convergence on Training Set

Ultimately, we want $z^{\mathsf{I}}_{i;\delta}(t)$ at the end of training, $t \to \infty$.

- Before we showed that for small enough η we have:

$$\lim_{t\to\infty} U(t) \propto \exp\Big(-\eta\widetilde{k}t\Big) \ .$$

Then, for training inputs, $z^{\mathsf{I}}_{i;\tilde{\alpha}}(t)$, will exponentially converge

$$\lim_{t\to\infty} z^{\mathsf{I}}_{i;\tilde{\alpha}}(t) = 0 \, .$$

- To see why this holds, note that we have

$$\lim_{s\to\infty} \mathbb{F}(s) \propto \exp\Big(-\eta\widetilde{k}s\Big) \ ,$$

since its leading behavior is linearly proportional to $\epsilon^{\mathsf{F}}_{j;\tilde{\alpha}}(t)$.

All together, this means that $z^{\mathsf{I}}_{i;\tilde{\alpha}}(t)$ converges as

$$\lim_{t\to\infty} z^{\mathsf{I}}_{i;\tilde{\alpha}}(t) \propto \lim_{t\to\infty} \Big\{\eta t \, \exp\Big[-(t-1)\eta\widetilde{k}\Big]\Big\} = 0 \, ,$$

slightly slower than the free solution $\epsilon^{\mathsf{F}}_{i;\tilde{\alpha}}(t) \propto \exp(-\widetilde{k}t)$.

Another Theoretical Minimum

$$z^{\mathsf{I}}_{i;\delta}(t) = \eta \sum_{s=0}^{t-1} \left[\mathbb{F}_{i;\delta}(s) - \sum_{\tilde{\alpha}_1,\tilde{\alpha}_2} k_{\delta\tilde{\alpha}_1} \widetilde{k}^{\tilde{\alpha}_1\tilde{\alpha}_2} \mathbb{F}_{i;\tilde{\alpha}_2}(s) \right] + \sum_{\tilde{\alpha}_1,\tilde{\alpha}_2} k_{\delta\tilde{\alpha}_1} \widetilde{k}^{\tilde{\alpha}_1\tilde{\alpha}_2} z^{\mathsf{I}}_{i;\tilde{\alpha}_2}(t)\,.$$

Another Theoretical Minimum

$$\lim_{t\to\infty} z^{\mathsf{I}}_{i;\delta}(t) = \left[\eta \sum_{s=0}^{\infty} \mathbb{F}_{i;\delta}(s)\right] - \sum_{\tilde{\alpha}_1,\tilde{\alpha}_2} k_{\delta\tilde{\alpha}_1} \widetilde{k}^{\tilde{\alpha}_1\tilde{\alpha}_2} \left[\eta \sum_{s=0}^{\infty} \mathbb{F}_{i;\tilde{\alpha}_2}(s)\right] .$$

Another Theoretical Minimum

$$\lim_{t\to\infty} z^{\mathsf{I}}_{i;\delta}(t) = \left[\eta \sum_{s=0}^{\infty} \mathbb{F}_{i;\delta}(s)\right] - \sum_{\tilde{\alpha}_1,\tilde{\alpha}_2} k_{\delta\tilde{\alpha}_1} \widetilde{k}^{\tilde{\alpha}_1\tilde{\alpha}_2} \left[\eta \sum_{s=0}^{\infty} \mathbb{F}_{i;\tilde{\alpha}_2}(s)\right] .$$

So we still have one more infinite sum to perform . . .

Another Theoretical Minimum

$$\lim_{t\to\infty} z^{\mathsf{I}}_{i;\delta}(t) = \left[\eta \sum_{s=0}^{\infty} \mathbb{F}_{i;\delta}(s)\right] - \sum_{\tilde{\alpha}_1,\tilde{\alpha}_2} k_{\delta\tilde{\alpha}_1} \widetilde{k}^{\tilde{\alpha}_1\tilde{\alpha}_2} \left[\eta \sum_{s=0}^{\infty} \mathbb{F}_{i;\tilde{\alpha}_2}(s)\right] .$$

So we still have one more infinite sum to perform . . .

$$\eta \sum_{s=0}^{\infty} \mathbb{F}_{i;\delta}(s) = -\sum_{\tilde{\alpha}} \left\{\eta \sum_{s=0}^{\infty} \left[k^{\mathsf{E}}_{\delta\tilde{\alpha}}(t) - k_{\delta\tilde{\alpha}}\right] \epsilon^{\mathsf{F}}_{i;\tilde{\alpha}}(t)\right\} + \frac{\eta}{2} \sum_{\tilde{\alpha}_1,\tilde{\alpha}_2} \mu_{\delta\tilde{\alpha}_1\tilde{\alpha}_2} \left\{\eta \sum_{s=0}^{\infty} \epsilon^{\mathsf{F}}_{i;\tilde{\alpha}_1}(t)\, \epsilon^{\mathsf{F}}_{i;\tilde{\alpha}_2}(t)\right\}$$

Another Theoretical Minimum

$$\lim_{t\to\infty} z^{\mathsf{I}}_{i;\delta}(t) = \left[\eta \sum_{s=0}^{\infty} \mathbb{F}_{i;\delta}(s)\right] - \sum_{\tilde{\alpha}_1,\tilde{\alpha}_2} k_{\delta\tilde{\alpha}_1} \widetilde{k}^{\tilde{\alpha}_1\tilde{\alpha}_2} \left[\eta \sum_{s=0}^{\infty} \mathbb{F}_{i;\tilde{\alpha}_2}(s)\right] .$$

So we still have one more infinite sum to perform . . .

$$\eta \sum_{s=0}^{\infty} \mathbb{F}_{i;\delta}(s) = \sum_{\tilde{\alpha}_0,\tilde{\alpha}} \left(\mu_{\delta_1\delta_2\tilde{\alpha}_0} + \mu_{\delta_2\delta_1\tilde{\alpha}_0}\right) \left\{\eta \sum_{s=0}^{\infty} a_{i;\tilde{\alpha}_0}(s)\, \epsilon^{\mathsf{F}}_{i;\tilde{\alpha}}(s)\right\}$$
$$+ \frac{\eta}{2} \sum_{\tilde{\alpha}_1,\tilde{\alpha}_2} \mu_{\delta\tilde{\alpha}_1\tilde{\alpha}_2} \left\{\eta \sum_{s=0}^{\infty} \epsilon^{\mathsf{F}}_{i;\tilde{\alpha}_1}(s)\, \epsilon^{\mathsf{F}}_{i;\tilde{\alpha}_2}(s)\right\}$$

Another *Aside*: Geometric Sums

$$\eta \sum_{s=0}^{\infty} \epsilon^{\mathsf{F}}_{i;\tilde{\alpha}_1}(t)\epsilon^{\mathsf{F}}_{i;\tilde{\alpha}_2}(t)$$

Another *Aside*: Geometric Sums

$$\eta \sum_{s=0}^{\infty} \epsilon^{\mathsf{F}}_{i;\tilde{\alpha}_1}(t)\epsilon^{\mathsf{F}}_{i;\tilde{\alpha}_2}(t)$$

$$= \eta \sum_{\tilde{\alpha}_3,\tilde{\alpha}_4} \sum_{s=0}^{\infty} \left[\left(\delta - \eta\widetilde{k}\right)^s\right]_{\tilde{\alpha}_1\tilde{\alpha}_3} \left[\left(\delta - \eta\widetilde{k}\right)^s\right]_{\tilde{\alpha}_2\tilde{\alpha}_4} \left(z_{i;\tilde{\alpha}_3} - y_{i;\tilde{\alpha}_3}\right)\left(z_{i;\tilde{\alpha}_4} - y_{i;\tilde{\alpha}_4}\right)$$

Another *Aside*: Geometric Sums

$$
\begin{aligned}
&\eta \sum_{s=0}^{\infty} \epsilon^{\mathsf{F}}_{i;\tilde{\alpha}_1}(t)\epsilon^{\mathsf{F}}_{i;\tilde{\alpha}_2}(t) \\
=&\eta \sum_{\tilde{\alpha}_3,\tilde{\alpha}_4} \sum_{s=0}^{\infty} \left[\left(\delta - \eta\widetilde{k}\right)^s\right]_{\tilde{\alpha}_1\tilde{\alpha}_3} \left[\left(\delta - \eta\widetilde{k}\right)^s\right]_{\tilde{\alpha}_2\tilde{\alpha}_4} \left(z_{i;\tilde{\alpha}_3} - y_{i;\tilde{\alpha}_3}\right)\left(z_{i;\tilde{\alpha}_4} - y_{i;\tilde{\alpha}_4}\right) \\
=&\sum_{\tilde{\alpha}_3,\tilde{\alpha}_4} X_{\mathsf{II}}^{\tilde{\alpha}_1\tilde{\alpha}_2\tilde{\alpha}_3\tilde{\alpha}_4} \left(z_{i;\tilde{\alpha}_3} - y_{i;\tilde{\alpha}_3}\right)\left(z_{i;\tilde{\alpha}_4} - y_{i;\tilde{\alpha}_4}\right)
\end{aligned}
$$

Another *Aside*: Geometric Sums

$$
\begin{aligned}
&\eta \sum_{s=0}^{\infty} \epsilon^{\mathsf{F}}_{i;\tilde{\alpha}_1}(t)\epsilon^{\mathsf{F}}_{i;\tilde{\alpha}_2}(t) \\
=&\eta \sum_{\tilde{\alpha}_3,\tilde{\alpha}_4} \sum_{s=0}^{\infty} \left[\left(\delta - \eta\widetilde{k}\right)^s\right]_{\tilde{\alpha}_1\tilde{\alpha}_3} \left[\left(\delta - \eta\widetilde{k}\right)^s\right]_{\tilde{\alpha}_2\tilde{\alpha}_4} \left(z_{i;\tilde{\alpha}_3} - y_{i;\tilde{\alpha}_3}\right)\left(z_{i;\tilde{\alpha}_4} - y_{i;\tilde{\alpha}_4}\right) \\
=& \sum_{\tilde{\alpha}_3,\tilde{\alpha}_4} X_{\mathsf{II}}^{\tilde{\alpha}_1\tilde{\alpha}_2\tilde{\alpha}_3\tilde{\alpha}_4} \left(z_{i;\tilde{\alpha}_3} - y_{i;\tilde{\alpha}_3}\right)\left(z_{i;\tilde{\alpha}_4} - y_{i;\tilde{\alpha}_4}\right)
\end{aligned}
$$

This last operation yielded an **inverting tensor** implicitly defined:

$$
\begin{aligned}
&\delta^{\tilde{\alpha}_1}_{\tilde{\alpha}_5}\delta^{\tilde{\alpha}_2}_{\tilde{\alpha}_6} \\
=& \sum_{\tilde{\alpha}_3,\tilde{\alpha}_4\in\mathcal{A}} X_{\mathsf{II}}^{\tilde{\alpha}_1\tilde{\alpha}_2\tilde{\alpha}_3\tilde{\alpha}_4} \frac{1}{\eta}\left[\delta_{\tilde{\alpha}_3\tilde{\alpha}_5}\delta_{\tilde{\alpha}_4\tilde{\alpha}_6} - (\delta_{\tilde{\alpha}_3\tilde{\alpha}_5} - \eta\widetilde{k}_{\tilde{\alpha}_3\tilde{\alpha}_5})(\delta_{\tilde{\alpha}_4\tilde{\alpha}_6} - \eta\widetilde{k}_{\tilde{\alpha}_4\tilde{\alpha}_6})\right] \\
=& \sum_{\tilde{\alpha}_3,\tilde{\alpha}_4\in\mathcal{A}} X_{\mathsf{II}}^{\tilde{\alpha}_1\tilde{\alpha}_2\tilde{\alpha}_3\tilde{\alpha}_4} \left(\widetilde{k}_{\tilde{\alpha}_3\tilde{\alpha}_5}\delta_{\tilde{\alpha}_4\tilde{\alpha}_6} + \delta_{\tilde{\alpha}_3\tilde{\alpha}_5}\widetilde{k}_{\tilde{\alpha}_4\tilde{\alpha}_6} - \eta\widetilde{k}_{\tilde{\alpha}_3\tilde{\alpha}_5}\widetilde{k}_{\tilde{\alpha}_4\tilde{\alpha}_6}\right) .
\end{aligned}
$$

Another Theoretical Minimum

$$
\begin{aligned}
& z_{i;\dot{\beta}}(\infty) \\
=& z_{i;\dot{\beta}}(0) - \sum_{\tilde{\alpha}_1,\tilde{\alpha}_2\in\mathcal{A}} k_{\dot{\beta}\tilde{\alpha}_1}\widetilde{k}^{\tilde{\alpha}_1\tilde{\alpha}_2}\epsilon^{\mathsf{F}}_{i;\tilde{\alpha}_2}(0) \\
& + \sum_{\tilde{\alpha}_1,\ldots,\tilde{\alpha}_4\in\mathcal{A}} \left[\mu_{\tilde{\alpha}_1\dot{\beta}\tilde{\alpha}_2} - \sum_{\tilde{\alpha}_5,\tilde{\alpha}_6\in\mathcal{A}} k_{\dot{\beta}\tilde{\alpha}_5}\widetilde{k}^{\tilde{\alpha}_5\tilde{\alpha}_6}\mu_{\tilde{\alpha}_1\tilde{\alpha}_6\tilde{\alpha}_2}\right] Z_{\mathsf{A}}^{\tilde{\alpha}_1\tilde{\alpha}_2\tilde{\alpha}_3\tilde{\alpha}_4}\epsilon^{\mathsf{F}}_{i;\tilde{\alpha}_3}(0)\,\epsilon^{\mathsf{F}}_{i;\tilde{\alpha}_4}(0) \\
& + \sum_{\tilde{\alpha}_1,\ldots,\tilde{\alpha}_4\in\mathcal{A}} \left[\mu_{\dot{\beta}\tilde{\alpha}_1\tilde{\alpha}_2} - \sum_{\tilde{\alpha}_5,\tilde{\alpha}_6\in\mathcal{A}} k_{\dot{\beta}\tilde{\alpha}_5}\widetilde{k}^{\tilde{\alpha}_5\tilde{\alpha}_6}\mu_{\tilde{\alpha}_6\tilde{\alpha}_1\tilde{\alpha}_2}\right] Z_{\mathsf{B}}^{\tilde{\alpha}_1\tilde{\alpha}_2\tilde{\alpha}_3\tilde{\alpha}_4}\epsilon^{\mathsf{F}}_{i;\tilde{\alpha}_3}(0)\,\epsilon^{\mathsf{F}}_{i;\tilde{\alpha}_4}(0)
\end{aligned}
$$

where the **algorithm projectors** are given by

$$
\begin{aligned}
Z_{\mathsf{A}}^{\tilde{\alpha}_1\tilde{\alpha}_2\tilde{\alpha}_3\tilde{\alpha}_4} \equiv& \widetilde{k}^{\tilde{\alpha}_1\tilde{\alpha}_3}\widetilde{k}^{\tilde{\alpha}_2\tilde{\alpha}_4} - \sum_{\tilde{\alpha}_5}\widetilde{k}^{\tilde{\alpha}_2\tilde{\alpha}_5}X_{\mathsf{II}}^{\tilde{\alpha}_1\tilde{\alpha}_5\tilde{\alpha}_3\tilde{\alpha}_4}\,, \\
Z_{\mathsf{B}}^{\tilde{\alpha}_1\tilde{\alpha}_2\tilde{\alpha}_3\tilde{\alpha}_4} \equiv& \widetilde{k}^{\tilde{\alpha}_1\tilde{\alpha}_3}\widetilde{k}^{\tilde{\alpha}_2\tilde{\alpha}_4} - \sum_{\tilde{\alpha}_5}\widetilde{k}^{\tilde{\alpha}_2\tilde{\alpha}_5}X_{\mathsf{II}}^{\tilde{\alpha}_1\tilde{\alpha}_5\tilde{\alpha}_3\tilde{\alpha}_4} + \frac{\eta}{2}X_{\mathsf{II}}^{\tilde{\alpha}_1\tilde{\alpha}_2\tilde{\alpha}_3\tilde{\alpha}_4}\,.
\end{aligned}
$$

Nearly-Kernel Methods

When the prediction of a *quadratic model* is computed in this way, call it a *nearly-kernel machine* or **nearly-kernel methods**.

Nearly-Kernel Methods

When the prediction of a *quadratic model* is computed in this way, call it a *nearly-kernel machine* or **nearly-kernel methods**.

Unlike *kernel methods*, this depends on the *learning algorithm*.

Nearly-Kernel Methods

When the prediction of a *quadratic model* is computed in this way, call it a *nearly-kernel machine* or **nearly-kernel methods**.

Unlike *kernel methods*, this depends on the *learning algorithm*.

- If we'd optimized by *direct optimization*, we'd have found:

$$Z_{\mathrm{A}}^{\tilde{\alpha}_1\tilde{\alpha}_2\tilde{\alpha}_3\tilde{\alpha}_4} = 0, \qquad Z_{\mathrm{B}}^{\tilde{\alpha}_1\tilde{\alpha}_2\tilde{\alpha}_3\tilde{\alpha}_4} = \frac{1}{2}\widetilde{k}^{\tilde{\alpha}_1\tilde{\alpha}_3}\widetilde{k}^{\tilde{\alpha}_2\tilde{\alpha}_4}.$$

Nearly-Kernel Methods

When the prediction of a *quadratic model* is computed in this way, call it a *nearly-kernel machine* or **nearly-kernel methods**.

Unlike *kernel methods*, this depends on the *learning algorithm*.

- If we'd optimized by *direct optimization*, we'd have found:

$$Z_{\mathsf{A}}^{\tilde{\alpha}_1\tilde{\alpha}_2\tilde{\alpha}_3\tilde{\alpha}_4} = 0, \qquad Z_{\mathsf{B}}^{\tilde{\alpha}_1\tilde{\alpha}_2\tilde{\alpha}_3\tilde{\alpha}_4} = \frac{1}{2}\widetilde{k}^{\tilde{\alpha}_1\tilde{\alpha}_3}\widetilde{k}^{\tilde{\alpha}_2\tilde{\alpha}_4}.$$

- In the ODE limit, we get different predictions

$$Z_{\mathsf{A}}^{\tilde{\alpha}_1\tilde{\alpha}_2\tilde{\alpha}_3\tilde{\alpha}_4} = Z_{\mathsf{B}}^{\tilde{\alpha}_1\tilde{\alpha}_2\tilde{\alpha}_3\tilde{\alpha}_4} \equiv \widetilde{k}^{\tilde{\alpha}_1\tilde{\alpha}_3}\widetilde{k}^{\tilde{\alpha}_2\tilde{\alpha}_4} - \sum_{\tilde{\alpha}_5}\widetilde{k}^{\tilde{\alpha}_2\tilde{\alpha}_5}X_{\mathsf{II}}^{\tilde{\alpha}_1\tilde{\alpha}_5\tilde{\alpha}_3\tilde{\alpha}_4},$$

$$\sum_{\tilde{\alpha}_3,\tilde{\alpha}_4\in\mathcal{A}} X_{\mathsf{II}}^{\tilde{\alpha}_1\tilde{\alpha}_2\tilde{\alpha}_3\tilde{\alpha}_4}\left(\widetilde{k}_{\tilde{\alpha}_3\tilde{\alpha}_5}\delta_{\tilde{\alpha}_4\tilde{\alpha}_6} + \delta_{\tilde{\alpha}_3\tilde{\alpha}_5}\widetilde{k}_{\tilde{\alpha}_4\tilde{\alpha}_6}\right) = \delta^{\tilde{\alpha}_1}_{\tilde{\alpha}_5}\delta^{\tilde{\alpha}_2}_{\tilde{\alpha}_6},$$

Nearly-Kernel Methods

When the prediction of a *quadratic model* is computed in this way, call it a *nearly-kernel machine* or **nearly-kernel methods**.

We again have two ways of thinking about the solution:

- we can use the optimal parameters to make predictions, or
- we can make *nearly-kernel predictions* in which the features, the meta features, and the model parameters do not appear.

Nearly-Kernel Methods

When the prediction of a *quadratic model* is computed in this way, call it a *nearly-kernel machine* or **nearly-kernel methods**.

We again have two ways of thinking about the solution:

- we can use the optimal parameters to make predictions, or
- we can make *nearly-kernel predictions* in which the features, the meta features, and the model parameters do not appear.

Predictions are made by direct comparison with the training set:

- It has the kernel linear piece $\propto y_{i;\tilde{\alpha}_2}$, and
- it also has a new quadratic piece $\propto y_{i;\tilde{\alpha}_1} y_{i;\tilde{\alpha}_2}$.

Algorithm Projectors

The prediction takes a **universal form**, with of the *algorithm dependence* of $\theta^\star$ is encoded in the *algorithm projectors*.

Algorithm Projectors

The prediction takes a **universal form**, with of the *algorithm dependence* of $\theta^\star$ is encoded in the *algorithm projectors*.

- Gives theoretical means of isolating the *inductive bias of the training algorithm* from all the other inductive biases.
- Essentially this is the sample-space *dual* of a learning algorithm, cf. the relationship between $\phi_j(x)$ and $k(x_{\delta_1}, x_{\delta_2})$.

Algorithm Projectors

The prediction takes a **universal form**, with of the *algorithm dependence* of $\theta^\star$ is encoded in the *algorithm projectors*.

- Gives theoretical means of isolating the *inductive bias of the training algorithm* from all the other inductive biases.
- Essentially this is the sample-space *dual* of a learning algorithm, cf. the relationship between $\phi_j(x)$ and $k(x_{\delta_1}, x_{\delta_2})$.

In principle, this gives a means of **inverse algorithm design**:

(i) engineer a desired functional form for the projectors

(ii) determine the parameter-space training algorithm that leads to the projectors having those desired properties or specific functional form.

Representation Learning

For simplicity, let's pick the **direct optimization** solution:

$$k^{\mathsf{E}}_{ii;\delta_1\delta_2}(\theta^\star) = k_{\delta_1\delta_2} + \sum_{\tilde{\alpha}_1,\tilde{\alpha}_2\in\mathcal{A}} (\mu_{\delta_1\delta_2\tilde{\alpha}_1} + \mu_{\delta_2\delta_1\tilde{\alpha}_1})\widetilde{k}^{\tilde{\alpha}_1\tilde{\alpha}_2} y_{i;\tilde{\alpha}_2} + O\Big(\epsilon^2\Big) \ .$$

Representation Learning

For simplicity, let's pick the **direct optimization** solution:

$$k^{\mathsf{E}}_{ii;\delta_1\delta_2}(\theta^\star) = k_{\delta_1\delta_2} + \sum_{\tilde{\alpha}_1,\tilde{\alpha}_2\in\mathcal{A}} (\mu_{\delta_1\delta_2\tilde{\alpha}_1} + \mu_{\delta_2\delta_1\tilde{\alpha}_1})\widetilde{k}^{\tilde{\alpha}_1\tilde{\alpha}_2} y_{i;\tilde{\alpha}_2} + O\left(\epsilon^2\right) \, .$$

Then, we can define a ***trained kernel*** that averages between the *fixed* kernel and *dynamical* effective kernels:

$$k^{\sharp}_{ii;\delta_1\delta_2} \equiv \frac{1}{2}\left[k_{\delta_1\delta_2} + k^{\mathsf{E}}_{ii;\delta_1\delta_2}(\theta^\star)\right] \, .$$

Representation Learning

For simplicity, let's pick the **direct optimization** solution:

$$k^{\mathsf{E}}_{ii;\delta_1\delta_2}(\theta^\star) = k_{\delta_1\delta_2} + \sum_{\tilde{\alpha}_1,\tilde{\alpha}_2\in\mathcal{A}} (\mu_{\delta_1\delta_2\tilde{\alpha}_1} + \mu_{\delta_2\delta_1\tilde{\alpha}_1})\widetilde{k}^{\tilde{\alpha}_1\tilde{\alpha}_2} y_{i;\tilde{\alpha}_2} + O\Big(\epsilon^2\Big) \ .$$

Then, we can define a ***trained kernel*** that averages between the *fixed* kernel and *dynamical* effective kernels:

$$k^{\sharp}_{ii;\delta_1\delta_2} \equiv \frac{1}{2}\Big[k_{\delta_1\delta_2} + k^{\mathsf{E}}_{ii;\delta_1\delta_2}(\theta^\star)\Big] \ .$$

Now the nearly-kernel prediction formula can be compressed,

$$z_i(x_{\dot{\beta}};\theta^\star) = \sum_{\tilde{\alpha}_1,\tilde{\alpha}_2\in\mathcal{A}} k^{\sharp}_{ii;\dot{\beta}\tilde{\alpha}_1} \widetilde{k^{\sharp}}^{\tilde{\alpha}_1\tilde{\alpha}_2}_{ii} y_{i;\tilde{\alpha}_2} + O\Big(\epsilon^2\Big) \ ,$$

taking the form of a *kernel prediction*, but with the benefit of nontrivial feature evolution incorporated into the trained kernel.

Representation Learning as Regularization

The *direct optimization* solution in parameter space is

$$z_i(x_{\dot{\beta}}; \theta^\star) = \sum_{j=0}^{n_f} W_{ij}^\star \phi_j(x_{\dot{\beta}}) + \frac{\epsilon}{2} \sum_{j_1, j_2=0}^{n_f} W_{ij_1}^\star W_{ij_2}^\star \psi_{j_1 j_2}(x_{\dot{\beta}})$$

and the optimal parameters can decompose as

$$W_{ij}^\star \equiv W_{ij}^{\mathsf{F}} + W_{ij}^{\mathsf{I}} \,,$$

where W_{ij}^{F} are the optimal parameters from the linear model.

Representation Learning as Regularization

The *direct optimization* solution in parameter space is

$$z_i(x_{\dot{\beta}}; \theta^\star) = \sum_{j=0}^{n_f} W_{ij}^\star \phi_j(x_{\dot{\beta}}) + \frac{\epsilon}{2} \sum_{j_1, j_2=0}^{n_f} W_{ij_1}^\star W_{ij_2}^\star \psi_{j_1 j_2}(x_{\dot{\beta}})$$

and the optimal parameters can decompose as

$$W_{ij}^\star \equiv W_{ij}^{\mathsf{F}} + W_{ij}^{\mathsf{I}} \,,$$

where W_{ij}^{F} are the optimal parameters from the linear model.

- The $O(\epsilon)$ tunings W_{ij}^{I} ruin the **fine tuning** of the W_{ij}^{F}, as they are constrained by the $\psi_{kj}(x)$ defined before training.

Representation Learning as Regularization

The *direct optimization* solution in parameter space is

$$z_i(x_{\dot{\beta}}; \theta^\star) = \sum_{j=0}^{n_f} W_{ij}^\star \phi_j(x_{\dot{\beta}}) + \frac{\epsilon}{2} \sum_{j_1, j_2=0}^{n_f} W_{ij_1}^\star W_{ij_2}^\star \psi_{j_1 j_2}(x_{\dot{\beta}})$$

and the optimal parameters can decompose as

$$W_{ij}^\star \equiv W_{ij}^{\mathrm{F}} + W_{ij}^{\mathrm{I}} \,,$$

where W_{ij}^{F} are the optimal parameters from the linear model.

- The $O(\epsilon)$ tunings W_{ij}^{I} ruin the **fine tuning** of the W_{ij}^{F}, as they are constrained by the $\psi_{kj}(x)$ defined before training.
- Assuming these $\psi_{kj}(x)$ are *useful*, we might expect that the quadratic model will overfit less and generalize better.

Course Plan

~~Lecture 1~~ ~~**Initialization, Linear Models**~~
- ~~§0 + §7.1 + §10.4~~

~~Lecture 2~~ ~~**Quadratic Models & Nearly-Kernel Methods**~~
- ~~§11.4 (+ §7.2) + §∞.2.2~~

Lecture 3 **The Principle of Sparsity (Recursing)**
- §4, §8, §11.2, §∞.3

Lecture 4 **The Principle of Criticality**
- §5, §9, §11.3, §∞.1, +§10.3

Lecture 5 **The End of Training & More**
- §∞.2.3 + Maybe { §A, §B, . . . }