A. Summary of Notations

D	input dimensions	P	number of random features
N	number of samples	$\gamma = {}^{P}/_{D}$	
$t = {}^{N}/_{D}$	training time, or equivalently rescaled number of training samples	y	true label
$\Omega^{\alpha} \in \mathbb{R}^{D\times D}$	covariance of the normal distribution of cluster α	$\frac{\mu^{\alpha}}{\sqrt{D}} \in \mathbb{R}^{D}$	mean of the Gaussian cluster α in the mixture
$x \in \mathbb{R}^{D}$	input	σ	standard deviation of the Gaussian clusters in a mixture with $\Omega_{\alpha} = \sigma^2\mathbb{I}$
$\mathsf{snr} = \frac{\lvert\mu\rvert}{\sigma\sqrt{D}}$	signal to noise ration	η	learning rate
κ	$L2$-regularisation constant	pmse	population mean squared error
ϵ_c	classification error		
$q(x\lvert y) = \sum_{\alpha\in\mathcal{S}(y)} P_{\alpha^{\pm}}\mathcal{N}_{\alpha}(x)$	conditional probability of x given the true label y		

Two layer neural networks (2LNN)

K	number of hidden nodes of the 2LNN	w_i^k	first layer weights
v^k	second layer weights	$g : \mathbb{R} \to \mathbb{R}$	activation function
$\lambda \equiv \sum_r w_r^k x_r/\sqrt{D}$	local field/pre-activation of the 2LNN	$\phi(\theta) = \sum_{k=1}^{K} v^k g(\lambda^k)$	output of the network
$Q_{\alpha}^{kl} = \sum_{rs} \frac{w_r^k \Omega_{rs}^{\alpha} w_s^l}{D}$	order parameter/ covariance of the local fields	$M_{\alpha}^{k} = \sum_r \frac{w_r^k \mu_r^{\alpha}}{D}$	order parameter/mean of the local fields
σ_0	weights are initialised i.i.d. from $\mathcal{N}(0, \sigma_0^2)$		

Random Features (RF)

$F \in \mathbb{R}^{P\times D}$	projection matrix $F_{ir} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1)$	$z \equiv \psi\left(\sum_{r=1}^{D} F_{ir}x_r/\sqrt{D}\right)$	features
$\psi : \mathbb{R} \to \mathbb{R}$	activation function applied element wise	$\phi(\theta) = \sum_{i=1}^{P} w_i z_i/\sqrt{D}$	output of the network
$\hat{W}$	fix point solution of the SGD update equation of RF		

B. Derivation of the dynamical equations

In this appendix, we derive the dynamical equations that describe the dynamics of two-layer neural networks trained on the Gaussian mixture from Sec. 2.2. We first derive a useful Lemma for the averages of weakly correlated random variables B.1, which we we then use in the derivation of the dynamical equations B.2

B.1. Moments of functions of weakly correlated variables

Here, we show how to compute expectation of functions of weakly correlated variables with non zero mean. The derivation follows the ones of (Goldt et al., 2020b) (see App. A). We extend their computations to include variables with non-zero means.

Consider the random variables x,$y \in \mathbb{R}$, jointly Gaussian with joint probability distribution:

$$P(x,y) = \frac{1}{2\pi\sqrt{\det M_2}} \exp\left[-\frac{1}{2}\begin{pmatrix} x-\bar{x} & y-\bar{y}\end{pmatrix} M_2^{-1} \begin{pmatrix} x-\bar{x} \\ y-\bar{y}\end{pmatrix}\right], \tag{B.1}$$

where we defined the mean of x, respectively y, as $\bar{x}$, respectively $\bar{y}$ and the covariance matrix:

$$M_2 = \begin{pmatrix} C_x & \epsilon M_{12} \\ \epsilon M_{12} & C_y \end{pmatrix}. \tag{B.2}$$

The weak correlation between x and y is encapsulated in the parameter $\epsilon \ll 1$ while $M_{12} \sim O(1)$.

We are interested in computing expectations of the form $\mathop{\mathbb{E}}_{(x,y)}[f(x)g(y)]$ with two real valued functions $f, g : \mathbb{R} \to \mathbb{R}$. Leveraging the weak correlation between x and y, we can expand the distribution Eq. (B.1) to linear order in ϵ, i.e.:

$$((\text{B.1})) = \frac{1}{2\pi\sqrt{C_x C_y}} \mathrm{e}^{-\frac{1}{2C_x}(x-\bar{x})^2 - \frac{1}{2C_y}(y-\bar{y})^2} \left[1 - \epsilon(x-\bar{x})\left(C_x^{-1}M_{12}C_y^{-1}\right)(y-\bar{y}) + O\left(\epsilon^2\right)\right] \tag{B.3}$$

Using the above, one can compute the expectations:

$$\begin{aligned}\mathop{\mathbb{E}}_{(x,y)}[f(x)g(y)] = &\mathop{\mathbb{E}}_{x}[f(x)] \mathop{\mathbb{E}}_{y}[g(y)] \\ &+ \epsilon \mathop{\mathbb{E}}_{x}[f(x)(x-\bar{x})]\left(C_x^{-1}M_{12}C_y^{-1}\right)\mathop{\mathbb{E}}_{y}[g(y)(y-\bar{y})] + O(\epsilon^2).\end{aligned} \tag{B.4}$$

The expectations are now taken over the 1-dimensional distributions of $x \sim \mathcal{N}(0, C_x)$ and $y \sim \mathcal{N}(0, C_x)$.

Similarly, consider the case of three weakly correlated real random variables x_i, $i = 1, 2, 3$ with mean $\bar{x}_i$ and covariance matrix M_3 such that

$$(M_3)_{ij} = \begin{cases} C_i, \text{ if } i = j \\ \epsilon M_{ij}, \text{ if } i \neq j \end{cases}. \tag{B.5}$$

One can use an expansion of the joint probability distribution of $\{x_i\}$ to linear order in ϵ to compute three point moments of real valued functions f, g, h as:

$$\begin{aligned}\mathop{\mathbb{E}}_{\{x\}_i}\left[f(x_1)g(x_2)h(x_3)\right] = &\mathop{\mathbb{E}}_{\{x\}_i}\left[f(x_1)\right]\left[g(x_2)\right]\left[h(x_3)\right] \\ &- \epsilon \mathop{\mathbb{E}}_{x_3}\left[h(x_3)\right] \mathop{\mathbb{E}}_{x_1}\left[(x_1-\bar{x}_1)f(x_1)\right]\left(C_x^{-1}M_{12}C_y^{-1}\right)\mathop{\mathbb{E}}_{x_2}\left[(x_2-\bar{x}_2)g(x_2)\right] \\ &- \epsilon \mathop{\mathbb{E}}_{x_2}\left[g(x_2)\right] \mathop{\mathbb{E}}_{x_1}\left[(x_1-\bar{x}_1)f(x_1)\right]\left(C_x^{-1}M_{13}C_y^{-1}\right)\mathop{\mathbb{E}}_{x_3}\left[(x_3-\bar{x}_3)h(x_3)\right] \\ &- \epsilon \mathop{\mathbb{E}}_{x_1}\left[f(x_1)\right] \mathop{\mathbb{E}}_{x_2}\left[(x_2-\bar{x}_2)g(x_2)\right]\left(C_x^{-1}M_{23}C_y^{-1}\right)\mathop{\mathbb{E}}_{x_3}\left[(x_3-\bar{x}_3)h(x_3)\right] + O(\epsilon^2).\end{aligned} \tag{B.6}$$

In the case in which x_1 and x_2 are weakly correlated with x_3 but not between each other, i.e. $\mathrm{Cov}(x_1, x_2) = M_{12} \sim O(1)$, one has:

$$\begin{aligned}\mathbb{E}\left[f(x_1)g(x_2)h(x_3)\right] = &\mathbb{E}\left[f(x_1)g(x_2)\right]\mathbb{E}\left[h(x_3)\right] \\ &+ \epsilon \frac{\mathbb{E}\left[h(x_3)(x_3-\bar{x}_3)\right]}{\left(C_{x_1}C_{x_2}-M_{12}^2\right)C_{x_3}}\left\{\mathbb{E}\left[f(x_1)g(x_2)(x_1-\bar{x}_1)\right]M_{13}C_{x_2}\right. \\ &\quad +\mathbb{E}\left[f(x_1)g(x_2)(x_2-\bar{x}_2)\right]M_{23}C_{x_1} \\ &\quad -\mathbb{E}\left[f(x_1)g(x_2)(x_1-\bar{x}_1)\right]M_{12}M_{23} \\ &\quad \left.-\mathbb{E}\left[f(x_1)g(x_2)(x_2-\bar{x}_2)\right]M_{13}M_{12}\right\} + O(\epsilon^2).\end{aligned} \tag{B.7}$$

B.2. Derivation of the ODEs

In this section, we derive the ODEs describing the dynamics of training of a 2LNN trained on inputs sampled from the distribution (6) with $L2$-regularisation constant κ. We restrict to the case where all the Gaussian clusters have the same covariance matrix, i.e. $\Omega^\alpha = \Omega \in \mathbb{R}^{D\times D}$.

In order to track the training dynamics, we analyse the evolution of the macroscopic operators defined in Eq. (9) allowing to compute the performances of the network at all training times.

At the sth step of training, the SGD update for the networks parameter is given by Eq. (7):

$$\mathrm{d}w_i^k \equiv \left(w_i^k\right)_{s+1} - \left(w_i^k\right)_s = -\frac{\eta_W}{\sqrt{D}}\left(v^k\Delta g'(\lambda^k)x_i + \kappa w_i^k\right), \qquad \mathrm{d}v^k = -\frac{\eta_v}{\sqrt{D}}\left(g(\lambda^k) + \kappa v^k\Delta\right). \tag{B.8}$$

In order to guarantee that the dynamics can be described by a set of ordinary equations in the $D\to\infty$ limit, we choose different scalings for the first and second layer learning rates:

$$\eta_W = \eta, \qquad \eta_v = \eta/D$$

for some constant η.

Update of the first layer weights To make progress, consider the eigen-decomposition of the covariance matrix:

$$\Omega_{rs} = \frac{1}{D}\sum_{\tau=1}^{D}\Gamma_{s\tau}\Gamma_{r\tau}\rho_\tau; \tag{B.9}$$

where we denote the eigenvalues as ρ_τ, their corresponding eigenvector as Γ_τ and the eigenvalue distribution as p_Ω. We further define the projection of the weights into the projected basis as

$$\tilde{w}_\tau^k \equiv \frac{1}{\sqrt{D}}\sum_{\tau=1}^{D}\Gamma_{s\tau}w_s^k \tag{B.10}$$

and similarly $\tilde{x}_\tau$ and $\tilde{\mu}_\tau^\alpha$ as the projected inputs and means. In this basis, the SGD update for the first layer weights is:

$$\mathrm{d}\tilde{w}_\tau^k \equiv \left(\tilde{w}_\tau^k\right)_{s+1} - \left(\tilde{w}_\tau^k\right)_s = -\frac{\eta}{\sqrt{D}}\left(v^k\Delta g'(\lambda^k)\tilde{x}_\tau + \kappa w_\tau^k\right). \tag{B.11}$$

The expectation of this update over the distribution Eq. (6) is given by:

$$\mathbb{E}\ \mathrm{d}\tilde{w}_\tau^k = \sum_{\alpha\in\mathcal{S}(+)} \mathcal{P}_\alpha\, \mathrm{d}\tilde{w}^k(\rho)_{\alpha^+} + \sum_{\alpha\in\mathcal{S}(-)} \mathcal{P}_\alpha\, \mathrm{d}\tilde{w}^k(\rho)_{\alpha^-}, \tag{B.12}$$

where we decomposed the expectation into the different clusters and introduced:

$$\mathrm{d}\tilde{w}^k(\rho)_{\alpha^\pm} = \pm\frac{\eta}{\sqrt{D}} v^k \mathcal{C}_\tau^k - \frac{\eta}{\sqrt{D}}\sum_{j\neq k} v^k v^j \mathcal{A}_\tau^{kj} - \frac{\eta}{\sqrt{D}} v^k v^k \mathcal{B}_\tau^k - \frac{\eta\kappa}{\sqrt{D}}\tilde{w}^k(\rho), \tag{B.13}$$

with the expectations $\mathcal{A}_\tau^{kj}, \mathcal{B}_\tau^k$ and $\mathcal{C}_\tau^k$ defined as:

$$\mathcal{A}_\tau^{kj} = \mathop{\mathbb{E}}_{\alpha} g'(\lambda^k) g(\lambda^j)\tilde{x}_\tau, \quad \mathcal{B}_\tau^k = \mathop{\mathbb{E}}_{\alpha} g'(\lambda^k) g(\lambda^k)\tilde{x}_\tau, \quad \mathcal{C}_\tau^k = \mathop{\mathbb{E}}_{\alpha} g'(\lambda^k)\tilde{x}_\tau. \tag{B.14}$$

A crucial observation, is that λ^k and the projected input $\tilde{x}_\tau$ are jointly Gaussian and weakly correlated, with a correlation of order $1/\sqrt{D}$:

$$\mathop{\mathbb{E}}_{\alpha} \lambda^k \tilde{x}_\tau = \frac{1}{\sqrt{D}}\rho_\tau \tilde{w}^k. \tag{B.15}$$

Thus, we can compute the expressions Eq. (B.14) using the proposition for weakly correlated variables derived in App. B.1. This gives:

$$\begin{aligned}
\mathcal{A}_\tau^{kj} &= \mathop{\mathbb{E}}_{\alpha} g'(\lambda^k) g(\lambda^j)\frac{\tilde{\mu}_\tau^\alpha}{\sqrt{D}} \\
&\quad + \frac{1}{Q^{kk}Q^{jj} - Q^{kj2}}\left\{\left(\mathop{\mathbb{E}}_{\alpha} g'(\lambda^k)\lambda^k g(\lambda^j) - \mathop{\mathbb{E}}_{\alpha} g'(\lambda^k) g(\lambda^j) M^{\alpha k}\right)\left(Q^{jj}\frac{\tilde{w}_\tau^k\rho_\tau}{\sqrt{D}} - Q^{kj}\frac{\tilde{w}_\tau^j\rho_\tau}{\sqrt{D}}\right)\right. \\
&\quad \left.\left(\mathop{\mathbb{E}}_{\alpha} g'(\lambda^k)\lambda^j g(\lambda^j) - \mathop{\mathbb{E}}_{\alpha} g'(\lambda^k) g(\lambda^j) M^{\alpha j}\right)\left(Q^{kk}\frac{\tilde{w}_\tau^j\rho_\tau}{\sqrt{D}} - Q^{kj}\frac{\tilde{w}_\tau^k\rho_\tau}{\sqrt{D}}\right)\right\} \\
\mathcal{B}_\tau^k &= \mathop{\mathbb{E}}_{\alpha} g'(\lambda^k) g(\lambda^k)\frac{\tilde{\mu}_\tau^\alpha}{\sqrt{D}} + \frac{1}{Q^{kk}}\left(\mathop{\mathbb{E}}_{\alpha} g'(\lambda^k)\lambda^k g(\lambda^k) - \mathop{\mathbb{E}}_{\alpha} g'(\lambda^k) g(\lambda^k) M^{\alpha k}\right)\frac{\tilde{w}_\tau^k\rho_\tau}{\sqrt{D}}, \\
\mathcal{C}_\tau^k &= \mathop{\mathbb{E}}_{\alpha} g'(\lambda^k)\frac{\tilde{\mu}_\tau^\alpha}{\sqrt{D}} + \frac{1}{Q^{kk}}\left(\mathop{\mathbb{E}}_{\alpha} g'(\lambda^k)\lambda^k - \mathop{\mathbb{E}}_{\alpha} g'(\lambda^k) M^{\alpha k}\right)\frac{\tilde{w}_\tau^k\rho_\tau}{\sqrt{D}},
\end{aligned} \tag{B.16}$$

where we used that the first moments of the local fields are given by the order parameters, $\mathbb{E}_\alpha[\lambda^k] = M^{\alpha k}$ and $\mathsf{Cov}(\lambda^k, \lambda^l) = Q^{kl}$. The multi-dimensional integrals of the activation function only depend on the order parameters at the previous step. We discuss how to obtain them, using monte-carlo methods in Sec 2.2. The averaged update of the first layer weights follows directly from Eq. (B.16).

Update of the Order parameters In order to derive the update equations for the order parameters, we introduce the densities $m(\rho, t)$ and $q(\rho, t)$. These depend on ρ and on the normalised number of steps $t = {}^{\mu}/_{D}$, which we interpret as a continuous time variable.

$$\begin{aligned}
m^{\alpha k}(\rho, t) &= \frac{1}{D\epsilon_\rho}\sum_\tau \tilde{w}_\tau^k\tilde{\mu}_\tau^\alpha \mathbb{1}_{(\rho_\tau\in[\rho,\rho+\varepsilon_\rho[)}, \\
q^{kl}(\rho, t) &= \frac{1}{D\varepsilon_\rho}\sum_\tau \tilde{w}_\tau^k\tilde{w}_\tau^l \mathbb{1}_{(\rho_\tau\in[\rho,\rho+\varepsilon_\rho[)},
\end{aligned} \tag{B.17}$$

where $\mathbb{1}(.)$ is the indicator function and the limit $\epsilon_\rho \to 0$ is taken after the thermodynamic limit. Using these definitions, the order parameters can be written as:

$$Q^{kl}(t) = \int \mathrm{d}\rho\ p_\Omega(\rho)\ \rho q^{kl}(\rho, t), \quad M^{\alpha k}(t) = \int \mathrm{d}\rho\ p_\Omega(\rho) m^{\alpha k}(\rho, t). \tag{B.18}$$

The equation of motion of m can can be easily computed using the update (B.13) and is given by:

$$\frac{\partial m^{\beta k}(\rho,t)}{\partial t} = \sum_{\alpha\in\mathcal{S}(+)} \mathcal{P}_\alpha \frac{\partial m^{\beta k}_{\alpha^+}(\rho,t)}{\partial t} + \sum_{\alpha\in\mathcal{S}(-)} \mathcal{P}_\alpha \frac{\partial m^{\beta k}_{\alpha^-}(\rho,t)}{\partial t}, \tag{B.19}$$

with

$$\begin{aligned}
\frac{\partial m^{\beta k}_{\alpha^\pm}(\rho,t)}{\partial t} =& \pm \eta v^k I_{31}(k) T^{\alpha\beta} \pm \frac{\eta\rho v^k}{Q^{kk}} \left(I_{32}(k,k) - I_{31}(k) M^{\alpha k}\right) m^{\beta k}(\rho,t) \\
&+ \sum_{j\neq k} \Big[-\eta v^k v^j I_{22}(k,j) T^{\alpha\beta} \\
&\quad + \frac{\eta\rho v^k v^j}{Q^{kk}Q^{jj} - Q^{kj2}} \Big\{ \left(-I_3(k,k,j) + I_{32}(k,j) M^{\alpha k}\right) \left(Q^{jj} m^{\beta k}(\rho,t) - Q^{kj} m^{\beta j}(\rho,t)\right) \\
&\qquad + \left(-I_3(k,j,j) + I_{32}(k,j) M^{\alpha j}\right) \left(Q^{kk} m^{\beta j}(\rho,t) - Q^{kj} m^{\beta k}(\rho,t)\right) \Big\} \Big] \\
&- \eta v^k v^k I_{22}(k,k) T^{\alpha\beta} - \frac{\eta\rho v^k v^k}{Q^{kk}} \left(I_3(k,k,k) - I_{22}(k,k) M^{\alpha k}\right) m^{\beta k}(\rho,t) - \kappa\eta m^{\beta k}(\rho,t).
\end{aligned} \tag{B.20}$$

Note how, in order to close the equation, we introduced an additional order parameter $T^{\alpha\beta} = \sum_{r=1}^{D} \frac{\mu_r^\alpha \mu_r^\beta}{D}$, which is entirely defined by the overlap of the means of the mixture under consideration and is therefore a constant of motion. For compactness, we defined the multidimensional integrals I of the activation function over the local fields as:

$$I_3(k,j,l) = \mathop{\mathbb{E}}_{\alpha} g'(\lambda^k)\lambda^k(\lambda^j) \tag{B.21a}$$

$$I_{32}(k,j) = \mathop{\mathbb{E}}_{\alpha} g'(\lambda^k)\lambda^j \tag{B.21b}$$

$$I_{31}(k) = \mathop{\mathbb{E}}_{\alpha} g'(\lambda^k) \tag{B.21c}$$

$$I_{22}(k,j) = \mathop{\mathbb{E}}_{\alpha} g'(\lambda^k) g(\lambda^j). \tag{B.21d}$$

The update of q can similarly be decomposed as a sum over the different Gaussian clusters:

$$\frac{\partial q^{kl}(\rho,t)}{\partial t} = \sum_{\alpha\in\mathcal{S}(+)} \mathcal{P}_\alpha \frac{\partial q^{kl}_{\alpha^+}(\rho,t)}{\partial t} + \sum_{\alpha\in\mathcal{S}(-)} \mathcal{P}_\alpha \frac{\partial q^{kl}_{\alpha^-}(\rho,t)}{\partial t} \tag{B.22}$$

The linear contribution to this update is directly computed by using Eq. (B.13) and is similar to the one for $m^{\beta k}(\rho,t)$. The quadratic contribution is obtained by using the fact that the projected inputs $\tilde{x}_\tau$ have a correlation of order $1/\sqrt{D}$ with the local fields. Therefore, to leading order, this contribution is given by terms of the form:

$$\frac{\eta^2}{D^2} \sum_\tau \rho_\tau \mathop{\mathbb{E}}_{\alpha} \left[g'(\lambda^k) g'(\lambda^l) g(\lambda^i) g(\lambda^j) \tilde{x}_\tau^2 \right] = \frac{\eta^2}{D} \mathop{\mathbb{E}}_{\alpha} \left[g'(\lambda^k) g'(\lambda^l) g(\lambda^i) g(\lambda^j) \right] \left(\sum_\tau \rho_\tau \mathop{\mathbb{E}}_{\alpha} \left[\tilde{x}_\tau^2 \right] \right) + O(D^{-3/2}) \tag{B.23}$$

Let us define the constant of motion $\chi^\alpha = \frac{1}{D}\sum_\tau \rho_\tau^2$, then the quadratic term in the update for $q_{\alpha^\pm}$ is given by:

$$\eta^2 \chi^\alpha v^k v^l \left(I_{42}(k,l) \mp 2 \sum_j v^j I_{43}(k,l,j) + \sum_{ja} v^j v^a I_4(k,l,j,a) \right). \tag{B.24}$$

The multidimensional integrals I are given by:

$$I_4(k,l,j,a) = \mathop{\mathbb{E}}_{\alpha} g'(\lambda^k) g'(\lambda^l) g(\lambda^j) g(\lambda^a) \quad I_{43}(k,l,j) = \mathop{\mathbb{E}}_{\alpha} g'(\lambda^k) g'(\lambda^l) g(\lambda^j) \quad I_{42}(k,l) = \mathop{\mathbb{E}}_{\alpha} g'(\lambda^k) g'(\lambda^l).$$

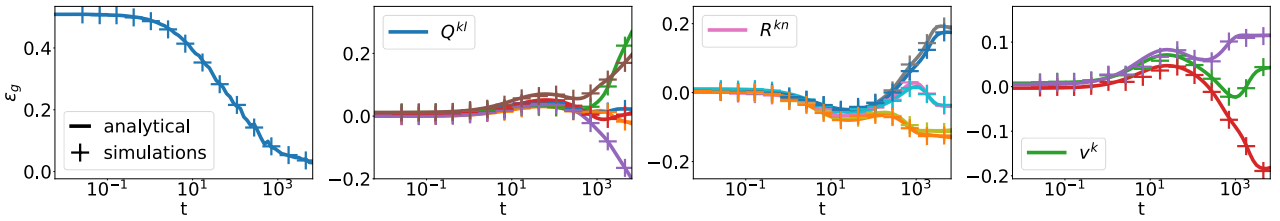

Figure 7. Agreement between simulations and ODEs when training a $K = 3$ 2LNN on a Gaussian mixture in $D = 800$ dimensions, with four Gaussian clusters with random covariance matrix Ω and random means (i.e. $\mu_r \sim \mathcal{N}(0, 1)$) for sigmoidal activation function. We verify that even at finite ($D = 800$) input dimension, the analytical prediction agree well with simulations. $\eta = 0.1$, $g(x) = \text{erf}\,(x/\sqrt{2})$, weights initialised with s.t.d. $\sigma_0 = 1$. Monte-Carlo integration performed with 10^{-4} samples.

Finally, the full equation of motion of q is written:

$$\begin{aligned}
\frac{\partial q^{kl}(\rho,t)_{\alpha^\pm}}{\partial t} &= \pm\,\eta v^k I_{31}(k) m^{\alpha k}(\rho,t) \pm \frac{\eta\rho v^k}{Q^{kk}}\left(I_{32}(k,k) - I_{31}(k)M^{\alpha k}\right) q^{kl}(\rho,t) \\
&+ \sum_{j\neq k}\Big[-\eta v^k v^j I_{22}(k,j) m^{\alpha k}(\rho,t) \\
&\quad + \frac{\eta\rho v^k v^j}{Q^{kk}Q^{jj} - Q^{kj2}}\Big\{\left(-I_3(k,k,j) + I_{32}(k,j)M^{\alpha k}\right)\left(Q^{jj}q^{kl}(\rho,t) - Q^{kj}q^{jl}(\rho,t)\right) \\
&\qquad + \left(-I_3(k,j,j) + I_{32}(k,j)M^{j\alpha}\right)\left(Q^{kk}q^{jl}(\rho,t) - Q^{kj}q^{kl}(\rho,t)\right)\Big\}\Big] \qquad \text{(B.25)} \\
&- \eta v^k v^k I_{22}(k,k) m^{\alpha k}(\rho,t) - \frac{\eta\rho v^k v^k}{Q^{kk}}\left(I_3(k,k,k) - I_{22}(k,k)M^{\alpha k}\right) q^{kl}(\rho,t) \\
&+ (k \leftrightarrow l) \\
&+ \eta^2 \chi^\alpha v^k v^l \left(I_{42}(k,l) \mp 2\sum_j v^j I_{43}(k,l,j) + \sum_{ja} v^j v^a I_4(k,l,j,a)\right) - 2\kappa\eta q^{kl}(\rho,t).
\end{aligned}$$

Update for the second layer weights The update of the second layer weights is also decomposed into the contribution of the different Gaussian clusters and follows from taking the expectation of Eq. (7b) on the GM distribution (6):

$$\begin{aligned}
\mathbb{E}\,\frac{\mathrm{d}v^k(t)}{\mathrm{d}t} &= \sum_{\alpha\in\mathcal{S}(+)} \mathcal{P}_\alpha \frac{\mathrm{d}v^k_{\alpha^+}(t)}{\mathrm{d}t} + \sum_{\alpha\in\mathcal{S}(-)} \mathcal{P}_\alpha \frac{\mathrm{d}v^k_{\alpha^-}(t)}{\mathrm{d}t}, \\
\frac{\mathrm{d}v^k_{\alpha^\pm}(t)}{\mathrm{d}t} &= \pm\,\eta \mathop{\mathbb{E}}_{\alpha} g(\lambda^k) - \eta\sum_j v^j \mathop{\mathbb{E}}_{\alpha} g(\lambda^k) g(\lambda^j) - \eta\kappa v^k
\end{aligned} \qquad \text{(B.26)}$$

Equations (B.20),(B.25) and (B.26) suffice to fully characterise the training dynamics, in the limit of high dimensions and online-learning, of a 2LNN trained on an arbitrary Gaussian mixture with $O(1)$ clusters each having mean μ^α and same covariance matrix Ω.

Agreement with Numerical Simulations Here, we verify the agreement of the ODEs derived above with simulation of 2LNN trained via online SGD.

To start, Fig. 7 displays the dynamics of a $K = 3$ network trained on a Gaussian mixture with 4 Gaussian clusters having covariance matrix $\Omega = {}^{F^T F}/\sqrt{D}$ and means μ^α, where the elements of both the matrix $F \in \mathbb{R}^{D\times D}$ and the means are sampled i.i.d. from a standard Gaussian distribution. The agreement between analytical prediction, given by integration of the ODEs, and simulations is very good both in the dynamics of the test error, of the order parameters and of the second layer weights.

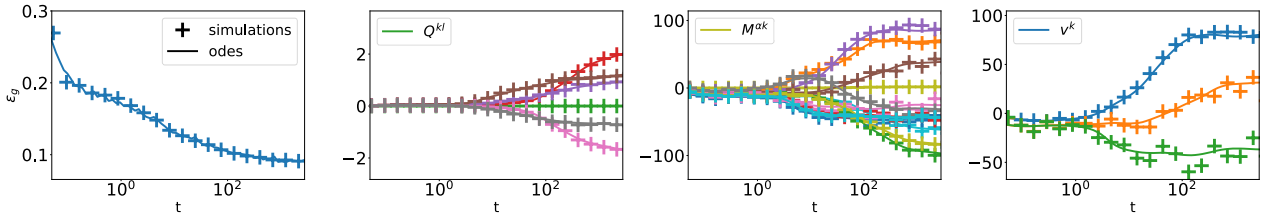

Figure 8. Agreement between simulations and ODEs when training a $K = 3$ 2LNN with sigmoidal activation function on a Gaussian mixture obtained from the FashionMNIST dataset. The analytical dynamics obtained by integrating the ODEs agrees well with the given by simulations. Thus, the ODEs provide a tool to study the importance of datastructure in training 2LNN. We leave this study for future work. $D = 784$, $\eta = 0.1$, $g(x) = \text{erf}\,(x/\sqrt{2})$, weights initialised with s.t.d. $\sigma_0 = 0.1$. Monte-Carlo integration performed with 10^4 samples.

Note that the equations of motion describe the evolution of the densities m and q averaged over the input distribution. The agreement between this evolution and simulations justifies, at posteriori, the implicit assumption that the stochastic part of the SGD increment (7) can be neglected in the $D \to \infty$ limit. We can thus conjecture that in the $D \to \infty$ limit, the stochastic process defined by the SGD updates converges to a deterministic process parametrised by the continuous time variables $t \equiv {}^{N}/_{D}$. We further add that the proof of this conjecture is not a straight-forward extension of the one of Goldt et al. (2019) for i.i.d. inputs since here, one must take into account the density of the covariance matrix.

The ODEs are valid for generic covariance matrix and means. Thus, they can be used to analyse the role of data structure in training 2LNNs. Although we leave a detailed analysis for future work, Fig. 8 gives an example of how this could be done in the case where a $K = 3$ 2LNN is trained on a GM obtained from the FashionMnist dataset. The GM is obtained by computing the means $\bar{x}^\alpha$ and covariance matrix $\mathsf{Cov}_\alpha(x, x)$ of each class in the dataset and assigning a label $+1$ or -1 to the different classes, as is commonly done in binary classification tasks. One could, for example, assign label $y = +1$ to the sneakers, boots, sandals, trousers and shorts categories and $y = -1$ to all others. Extending our analysis to C-class classification is straight forward and follows the analysis of Yoshida et al. (2019). The inputs are then sampled from a GM where the cluster's mean are given by $\bar{x}^\alpha$ and the covariance matrix Ω is the mean covariance of all classes: $\Omega = {}^{1}/_{\text{n}_\text{classes}} \sum_\alpha \mathsf{Cov}_\alpha(x, x)$. Note the similarity between this procedure and *linear discriminant analysis* commonly used in statistics. The agreement between simulations and analytical predictions is again very good, both at the level of the test error and of the order parameters.

B.3. Simplified ansatz to solve the ODEs for the XOR-like mixture

Here, we detail the procedure, introduced in Sec. 2.3, used to find the long time $t \to \infty$ performance of 2LNN by making an *ansatz* on the form of the order parameters that solve the fix point equations. The motivation for doing so, as argued in of the main text, is that integrating the ODEs is numerically expensive as it requires evaluating various multidimensional integrals and the number of equations to integrate scales as K^2. In order to extract information about the asymptotic performances of the network, one can look for a fix point of the ODEs. However, the number of coupled equations to be solved, also scales quadraticaly with K and is already 26 for a $K = 4$ student. The trick is to make an *ansatz*, with fewer degrees of freedom, on the order parameters that solve the equations. Used in this way, the ODEs have generated a wealth of analytical insights into the dynamics and the performance of 2LNN in the classical teacher-student setup (Biehl & Schwarze, 1995; Saad & Solla, 1995b;b; Biehl et al., 1996; Saad, 2009; Yoshida & Okada, 2019; Yoshida et al., 2019; Goldt et al., 2020b). In all these works though, an important simplification occured because the means of the local fields were all zero by construction. This simplification allowed the fixed points to be found analytically in some cases. Here, the means of the local fields evaluated over individual Gaussians in the mixture are *not* zero, so we have to resort to numerical means to find the fixed points of the ODEs.

Consider, for example, a 2LNN trained on the XOR-like mixture of Fig. 1. The Gaussian clusters have covariance $\Omega = \sigma^2 \mathbb{I}$ and means chosen as in the left-hand-side diagram of Fig. 1, with the remaining $D-2$ components set to 0. This configuration leads to the constrain $w^k \cdot \mu^{\pm 0} = -w^k \cdot \mu^{\pm 1}$ that, in terms of overlap matrices, forces $M^{\alpha k} = -M^{\alpha+1 k}$ thus halving the number of free parameters in M. It is also clear that the only components of the weight vectors which contribute to the error

are those in the plane spanned by the means of the mixture. The additional $D-2$ components can be taken to 0: i.e. $w_r^k = 0$ for $r = 2, .., D$. This condition allows to decompose the weight vectors as:

$$w_s^k = \sum_r \frac{w_r^k \mu_r^{+0}}{\sqrt{D}} \frac{\mu_s^{+0}}{\sqrt{D}} + \sum_r \frac{w_r^k \mu_r^{-0}}{\sqrt{D}} \frac{\mu_s^{-0}}{\sqrt{D}} \tag{B.27}$$

This decomposition, fully constrains the overlap matrix Q^{kl} in terms of $M^{\alpha k}$:

$$\begin{aligned} Q^{kl} &= \sigma^2 \sum_s \frac{w_s^k w_s^l}{D} \\ &= \sigma^2 \left(\sum_r \frac{w^k \mu_r^{+0}}{D} \right) \left(\sum_a \frac{w_a^l \mu_a^{+0}}{D} \right) \frac{|\mu^{+0}|^2}{D} + \sigma^2 \left(\sum_r \frac{w_r^k \mu_r^{-0}}{D} \right) \left(\sum_a \frac{w_a^l \mu_a^{-0}}{D} \right) \frac{|\mu^{-0}|^2}{D} \\ \implies Q^{kl} &= \sigma^2 \left(M^{+0k} M^{+0l} + M^{-0k} M^{-0l} \right), \end{aligned} \tag{B.28}$$

where we used that in the XOR-like mixture, $|\mu^\alpha| = \sqrt{D}$ and $\mu^{0+} \cdot \mu^{0-} = 0$. From the symmetry between the positive and negative sign clusters of the mixture, in the fix point configuration, for every weight having norm $|w|$ and at an angle α with the mean of a positive cluster, there is a corresponding weight of the same norm, at an angle α with a negative mean. I.e. the angles of the weight vectors w^k to the means μ^α, as well as the norms of the weights, are 2×2 equal (one for the positive sign cluster and the other for the negative sign one). This constrains further half the number of free parameters in the overlap matrix M, which are down to $4K/(2 \times 2) = K$. The second layer weights v are fully constrained by requiring the output of the student to be ± 1 when evaluated on the means. Putting everything together, one is left with K equations to solve for the $K/2$ angles and the $K/2$ norms, or equivalently, for the K free parameters in the overlap matrix M. The agreement between the solution found by solving this reduced set of equations and simulations is displayed both in Fig. 3, where we use it to predict the evolution of the test error with the $L2-$regularisation constant.

C. Transforming a Gaussian mixture with random features

C.1. The distribution of random features is still a mixture

Given an input $x = (x_i) \in \mathbb{R}^D$ sampled from the distribution (6), we consider the *feature* vector $z = (z_i) \in \mathbb{R}^N$

$$z_i = \psi(u_i), \qquad u_i \equiv \sum_{r=1}^{D} \frac{1}{\sqrt{D}} F_{ir} x_r \tag{C.1}$$

where $F \in \mathbb{R}^{D \times P}$ is a random projection matrix and $\psi : \mathbb{R} \to \mathbb{R}$ is an element-wise non linearity. The distribution of z can be computed as:

$$\begin{aligned} p_z(z) &= \int_{\mathbb{R}^D} dx p_x(x) \delta \left(z - \psi \left(\frac{xF}{\sqrt{D}} \right) \right) \\ &= \int_{\mathbb{R}^D} dx \sum_y q(y) q(x|y) \delta \left(z - \psi \left(\frac{xF}{\sqrt{D}} \right) \right) \\ &\equiv \sum_y q(y) \sum_{\alpha \in \mathcal{S}(y)} \mathcal{P}_\alpha p_z^\alpha(z), \end{aligned} \tag{C.2}$$

with

$$p_z^\alpha(z) = \int_{\mathbb{R}^D} dx \delta \left(z - \psi \left(\frac{xF}{\sqrt{D}} \right) \right) \mathcal{N}_\alpha \left(\frac{\mu^\alpha}{\sqrt{D}}, \Omega^\alpha \right) \tag{C.3}$$

Crucially, the distribution of the features z is still a mixture of distributions. We can thus restrict to studying the transformation of a Gaussian random variable

$$x_r = \frac{\mu_r}{\sqrt{D}} + \sigma w_r \tag{C.4}$$

where w_r is a standard Gaussian. The scaling of μ_r and σ is chosen according to which regime (low or high snr) one chooses to study. We aim at computing the distribution, in particular the two first moments, of the feature z defined in Eq. (C.1). By

construction, the random variables u_i are Gaussian with first two moments:

$$\mathbb{E}\, u_i = \frac{1}{\sqrt{D}}\tilde{\mu}_i, \qquad \tilde{\mu}_i \equiv \sum_r \frac{F_{ir}\mu_r}{\sqrt{D}} \tag{C.5}$$

$$\mathbb{E}\, u_i u_j = \frac{1}{D}\tilde{\mu}_i\tilde{\mu}_j + \sigma^2 \sum_r \frac{F_{ir}F_{jr}}{D} \tag{C.6}$$

C.2. Low signal-to-noise ratio

Here we compute the statistics of the features, for general activation function, in the low signal to noise regime, for which $|\mu|/\sqrt{D} \sim O(1)$ and $\sigma \sim O(1)$ so that the Gaussian clusters are a distance of order 1 away from the origin.

The mean of z_i can be written as:

$$\mathbb{E}\, z_i = \mathbb{E}\, \psi(u_i) = \mathbb{E}\, \psi\left(\frac{\tilde{\mu}_i}{\sqrt{D}} + \sigma\zeta\right), \tag{C.7}$$

where $\zeta \in \mathbb{R}$ is a standard Gaussian variable. In the scaling we work in, where P and D are send to infinity with their ratio fixed, and $|\mu|/\sqrt{D} \sim O(1)$, $\tilde{\mu}_i/\sqrt{D}$ is of order $O(1/\sqrt{D})$. Thus, the activation function ψ can be expanded around $\sigma\zeta$:

$$\mathbb{E}\, z_i = \mathbb{E}\, \psi\left(\sigma\zeta\right) + \frac{\tilde{\mu}_i}{\sigma\sqrt{D}}\mathbb{E}\, \zeta\psi\left(\sigma\zeta\right) + O\left(\frac{1}{D}\right). \tag{C.8}$$

where we used integration by part to find $\sigma\mathbb{E}\, \partial\psi\left(\sigma\zeta\right) = \mathbb{E}\, \zeta\psi\left(\sigma\zeta\right)$.

For the covariance matrix, we separate the computation of the diagonal from the off-diagonal. Starting with the diagonal elements:

$$\begin{aligned}\mathbb{E}\left[z_i^2\right] =& \mathbb{E}\left[\left(\psi\left(\sigma\zeta\right) + \frac{\tilde{\mu}_i}{\sigma\sqrt{D}}\partial\psi\left(\sigma\zeta\right)\right)^2\right] \\ =& \mathbb{E}\left[\psi\left(\sigma\zeta\right)^2\right] + \frac{\tilde{\mu}_i}{\sigma\sqrt{D}}\mathbb{E}\, \zeta\psi^2\left(\sigma\zeta\right) + O\left(\frac{1}{D}\right)\end{aligned} \tag{C.9}$$

where, once gain, integration by parts was used to obtain $2\sigma\mathbb{E}\left[\psi\left(\sigma\zeta\right)\partial\psi\left(\sigma\zeta\right)\right] = \mathbb{E}\, \zeta\psi^2\left(\sigma\zeta\right)$.

In order to compute the off-diagonal elements, we note that different components of u are weakly correlated since $\mathrm{Cov}(u_i, u_j) = \sigma^2 \sum_r F_{ir}F_{jr}/D \sim O(1/\sqrt{D})$. We can therefore apply formula Eq. (B.4) for weakly correlated variables:

$$\begin{aligned}\mathop{\mathbb{E}}_{(z_i,z_j)} z_i z_j =& \mathop{\mathbb{E}}_{(u_i,u_j)} \psi(u_i)\psi(u_j) \\ =& \mathop{\mathbb{E}}_{u_i} \psi(u_i) \mathop{\mathbb{E}}_{u_j} \psi(u_j) + \frac{1}{\sigma^2}\sum_r \frac{F_{ir}F_{jr}}{D} \mathop{\mathbb{E}}_{u_i} u_i\psi(u_i) \mathop{\mathbb{E}}_{u_j} u_j\psi(u_j) + O(\frac{1}{D})\end{aligned} \tag{C.10}$$

where the averages are now over the one dimensional distributions of $u_i \sim \mathcal{N}\left(\frac{\tilde{\mu}_i}{\sqrt{D}}, \sigma^2\right)$. We can now replace in the above $u_i = \frac{\tilde{\mu}_i}{\sqrt{D}} + \sigma\zeta$ and keep only leading order terms:

$$\begin{aligned}\mathsf{Cov}(z_i, z_j) &= \sum_r \frac{F_{ir}F_{jr}}{\sigma^2 D}\left(\psi(\sigma\zeta + \frac{\mu_i}{\sqrt{D}})(\sigma\zeta + \frac{\mu_i}{\sqrt{D}})\right)\left(\psi(\sigma\zeta + \frac{\mu_j}{\sqrt{D}})(\sigma\zeta + \frac{\mu_j}{\sqrt{D}})\right) \\ &= \sum_r \frac{F_{ir}F_{jr}}{\sigma^2 D}\left(\psi(\sigma\zeta)\sigma\zeta + O(\frac{1}{\sqrt{D}})\right)\left(\psi(\sigma\zeta)\sigma\zeta + O(\frac{1}{\sqrt{D}})\right) \\ &= \mathbb{E}\left[\zeta\psi(\sigma\zeta)\right]^2 \sum_r \frac{F_{ir}F_{jr}}{D} + O\left(\frac{1}{D}\right) \quad \text{, for } i \neq j.\end{aligned} \tag{C.11}$$

Thus yielding the final result:

$$\mathsf{Cov}(z_i, z_j) = \mathbb{E}\left[\zeta\psi(\sigma\zeta)\right]^2 \sum_r \frac{F_{ir}F_{jr}}{D} \quad \text{, for } i \neq j. \tag{C.12}$$

We define the constants a, b and c as in Eq. (**??**) and d as:

$$a = \mathbb{E}\,\psi(\sigma\zeta)\,, \quad b = \mathbb{E}\,\zeta\psi(\sigma\zeta)\,, \quad c^2 = \mathbb{E}\left[\psi(\sigma\zeta)^2\right], \quad d^2 = \mathbb{E}\,\zeta\psi^2(\sigma\zeta) \tag{C.13}$$

These definitions together with Eq. (C.8), Eq. (C.9) and Eq. (C.12) lead to the statistics of Eq. (18) and Eq. (19):

$$\begin{aligned}
\mathbb{E}\,z_i &= a + \frac{\tilde{\mu}_i}{\sigma\sqrt{D}} b \\
\mathsf{Cov}(z_i, z_j) &= \begin{cases} c^2 - a^2 + \underbrace{\frac{\tilde{\mu}_i}{\sigma\sqrt{D}}\left(d^2 - 2ab\right)}_{O(\frac{1}{\sqrt{D}})\text{: subleading if } c^2 - a^2 > 0} + O(\frac{1}{D}) & \text{, if } i = j \\ b^2 \sum_r \frac{F_{ir}F_{jr}}{D} + O(\frac{1}{D}) & \text{, if } i \neq j \end{cases}
\end{aligned} \tag{C.14}$$

The above shows that, the transformation of the means is only linear and in the low snr regime, the XOR-like mixture of Fig. 1, is transformed into a XOR-like mixture in feature space which cannot be learned by linear regression. Note, that the performance of linear regression on the features is equivalent to its performance on inputs $\tilde{z} \in \mathbb{R}^P$ sampled from a Gaussian equivalent model defined as:

$$\tilde{z}_i = \mathbb{E}\,z_i + \sum_j \Omega_{ij}^{1/2} \zeta_j, \tag{C.15}$$

where $\mathsf{Cov}(z_i, z_j) \equiv \sum_k \Omega_{ik}^{1/2}\Omega_{jk}^{1/2}$ and $\zeta \in \mathbb{R}^P$ is a random vector with components sampled i.i.d. from a standard Gaussian distribution.

C.3. ReLU features

In the case of Relu activation function i.e. $\psi(x) = \max(0, x)$, the mean and the covariance of the features can be evaluated analytically for all snr regimes. The distribution of the features within each cluster is given by a modified Gaussian: the probability mass of the Gaussian on the negative real axis is concentrated at the origin while the distribution on the positive axis is unchanged.

In particular, the integral to obtain the mean of z can be computed analytically and is given by:

$$\mathbb{E}\,z_i = \mathbb{E}\;\mathrm{ReLU}(u_i) = \sigma\left[\frac{\tilde{\rho}_i}{2}\left(1 + \mathrm{erf}(\frac{\tilde{\rho}_i}{\sqrt{2}})\right) + \frac{1}{\sqrt{2\pi}} e^{-\tilde{\rho}_i^2/2}\right], \tag{C.16}$$

where we defined:

$$\tilde{\rho}_i \equiv \frac{\tilde{\mu}_i}{\sqrt{D}\sigma} = \frac{\sum_r F_{ir}\mu_r}{D\sigma}.$$

The covariance is once again computed by separating the diagonal terms from the off-diagonal ones. The integral to obtain the diagonal terms has an analytical expression found to be:

$$\mathsf{Cov}(z_i, z_i) = \mathbb{E}\left[\mathrm{ReLU}(u_i)^2\right] - \mathbb{E}\left[\mathrm{ReLU}(u_i)\right]^2 = \sigma^2\left[\frac{\tilde{\rho}_i}{\sqrt{2\pi}} e^{-\tilde{\rho}_i^2/2} + \frac{1}{2}(\tilde{\rho}_i^2 + 1)(1 + \mathrm{erf}(\frac{\tilde{\rho}_i}{\sqrt{2}})) - (\mathbb{E}\,z_i)^2\right] \tag{C.17}$$

For the off-diagonal components, we again use that the covariance of the different components of the u_i is of order $1/\sqrt{D}$ as $\mathsf{Cov}(u_i, u_j) = \sigma^2 \sum_r F_{ir}F_{jr}/D$. Therefore, to evaluate $\mathbb{E}\left[\mathrm{ReLU}(u_i)\,\mathrm{ReLU}(u_j)\right]$, we can use the result Eq. (B.4) for weakly correlated variables with $\epsilon M_{12} = \sigma^2 \sum_r F_{ir}F_{jr}/D$. Then to leading order in $\frac{1}{D}$, one finds:

$$\begin{aligned}
\mathrm{Cov}(z_i, z_j) &= \mathbb{E}\left[\mathrm{ReLU}(u_i)\,\mathrm{ReLU}(u_j)\right] - \mathbb{E}\left[\mathrm{ReLU}(u_i)\right]\mathbb{E}\left[\mathrm{ReLU}(u_j)\right] \\
&= \sum_r \frac{F_{ir}F_{jr}}{\sigma^2 D}\mathbb{E}\left[u_i\,\mathrm{ReLU}(u_i)\right]\mathbb{E}\left[u_j\,\mathrm{ReLU}(u_j)\right] + O(\frac{1}{D})
\end{aligned} \tag{C.18}$$

where the expectations above are over one dimensional distributions $u_i \sim \mathcal{N}\left(\sum_r \frac{F_{ir}\mu_r}{D}, \sigma^2\right)$. The integrals have an analytical closed form expression, which yields the final result for the covariance:

$$\mathrm{Cov}(z_i, z_j) = \begin{cases} \sigma^2\left[\frac{\tilde{\rho}_i}{\sqrt{2\pi}} e^{-\tilde{\rho}_i^2/2} + \frac{1}{2}(\tilde{\rho}_i^2 + 1)(1 + \mathrm{erf}(\frac{\tilde{\rho}_i}{\sqrt{2}})) - (\mathbb{E}\,z_i)^2\right] & \text{if } i = j, \\ \frac{\sigma^2}{4}\frac{\sum_r F_{ir}F_{jr}}{D}\left(1 + \mathrm{erf}(\frac{\tilde{\rho}_i}{\sqrt{2}})\right)\left(1 + \mathrm{erf}(\frac{\tilde{\rho}_j}{\sqrt{2}})\right) + O(\frac{1}{D}) & \text{if } i \neq j \end{cases} \tag{C.19}$$

C.4. Relation with the kernel

As discussed in Sec. 3 of the main text , the performances of kernel methods can be studied by using the convergence of RF to a kernel in the $\gamma \to \infty$ limit taken after the $D, P \to \infty$ limit (Rahimi & Recht, 2008):

$$K(x,y) = \frac{1}{P}\sum_{i=1}^{P} \mathop{\mathbb{E}}_{F}\left[\psi\left(\sum_{r=1}^{D}\frac{x_r F_{ir}}{\sqrt{D}}\right)\psi\left(\sum_{s=1}^{D}\frac{y_s F_{is}}{\sqrt{D}}\right)\right],$$

In the low snr regime where $|\mu|/\sqrt{D} \sim O(1)$, the action of the kernel is essentially linear. The three constants a, b and c, defined in Eq. (**??**), can be expressed equivalently in terms of the activation function ψ or of the kernel. Consider ω_1, $\omega_2 \in \mathbb{R}$, two i.i.d. standard Gaussian random variables, and denote by angle brackets $\langle\cdot\rangle$ the expectation over w_1, w_2. Then, by the definition $K(x,y)$ one has:

$$\langle K(\sigma\omega_1, \sigma\omega_1)\rangle = \frac{1}{P}\sum_{i=1}^{P}\langle\psi(u_i)^2\rangle = c^2 \tag{C.20}$$

where $u \in \mathbb{R}^P$ is the random vector whose moments are defined in Eq. (C.5) and we used the element wise convergence of $\psi(\sum_r F_{ir}x_r)\psi(\sum_r F_{is}y_s)/P$ to its expected value (Rahimi & Recht, 2008). Similarly, one has:

$$\langle K(\sigma\omega_1, \sigma\omega_2)\rangle = \frac{1}{P}\sum_{i=1}^{P}\langle\psi(u_i)\rangle^2 = a^2 \tag{C.21}$$

Finally, for b one has to perform a linear expansion of the kernel around the noise variable $\sigma\omega_1$:

$$\begin{aligned}
\langle K(\sigma\omega_1 + \frac{\mu}{\sqrt{D}}, \sigma\omega_2 + \frac{\mu}{\sqrt{D}})\rangle =& \frac{1}{P}\sum_{i=1}^{P}\langle \mathop{\mathbb{E}}_{F}\left[\psi(u_{i1} + \sum_r \frac{\mu_r F_{ir}}{D})\rangle\langle\psi(u_{i2} + \sum_r \frac{\mu_r F_{ir}}{D})\right]\rangle \\
=& \frac{1}{P}\sum_{i=1}^{P}\langle\psi(u_i)\rangle^2 + \sum_{rs}\frac{\mu_r\mu_s}{D^2}\sum_{i=1}^{P}\mathop{\mathbb{E}}_{F}\frac{F_{ir}F_{is}}{P}\langle\psi'(u_i)\rangle^2 \\
=& a^2 + \frac{1}{D\sigma^2}b^2, \\
\implies b^2 =& D\sigma^2\left(-a^2 + \langle K(\sigma\omega_1 + \frac{\mu}{\sqrt{D}}, \sigma\omega_2 + \frac{\mu}{\sqrt{D}})\rangle\right)
\end{aligned} \tag{C.22}$$

These expressions allow to express the statistical properties of the features z, and to asses the performance of RF and kernel methods, directly in terms of the kernel without requiring the explicit form of the activation function.

For completeness, we give the analytical expression of the kernel corresponding to ReLU random features, i.e. $\psi(x) = \max(0, x)$:

$$K_{\text{ReLU}}(x,y) = \frac{|x||y|}{8\pi D}\left\{2|\sin(\theta)| + \cos(\theta)\left(\pi + 2\,\text{Arctan}\left(\frac{\cos(\theta)}{|\sin(\theta)|}\right)\right)\right\}, \tag{C.23}$$

where we defined the angle θ between the two vectors $x, y \in \mathbb{R}^D$ such that $|x|, |y| > 0$:

$$\theta = \frac{x\cdot y}{|x||y|} \tag{C.24}$$

From Eq. (C.23), one sees that in case of ReLU activation function, the kernel is an angular kernel, i.e. it depends on the angle between x and y.

D. Final test error of random features

This section details the computations leading to Eq. (14) and Eq. (16) allowing to obtain the asymptotic performances of RF trained via online SGD on a mixture of Gaussian distribution.

Applying random features on $x \in \mathbb{R}^D$ sampled from the distribution (6) is equivalent to performing linear regression on the features $z \in \mathbb{R}^P$ with covariance Ω^z and mean μ_z. In the following, for clarity, we assume the features are centred, so

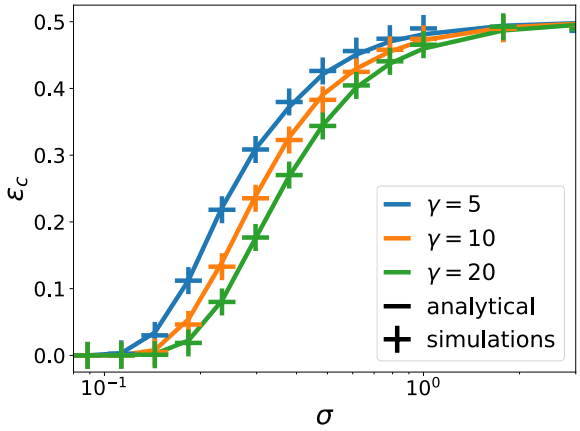

Figure 9. Agreement between analytical predictions and simulations of the classification error of RF trained on the XOR like mixture of Fig. 1 for increasing σ and various γ. The analytical predictions are obtained by computing the moments of the features z using Eq. C.16 and Eq. C.19. These are then used to obtain, first the asymptotic solution of RF via Eq. D.6, which is in turn plugged in Eq. D.9. The final result is then given by Eq. D.8. Simulation results are obtained by training online an RF network with $\eta = 0.1$ until convergence. *Parameters:* $D = 800$, $\eta = 0.1$, $P = \gamma D$, $|\mu|/\sqrt{D} = 1$.

that $\mu_z = 0$, extending the computation to non centred features is straight-forward. The means of the individual clusters are however non zero. The output of the network at a step t in training is given by $\phi(w)_t = \sum_{i=1}^{P} w_i z_{ti}/\sqrt{P}$. We train the network in the online limit by minimising the squared loss between the output of the network and the label y_t. The pmse is given by:

$$\mathsf{pmse}(w) = \frac{1}{2}\mathbb{E}\left(y - \sum_{i=1}^{P}\frac{w_i z_i}{\sqrt{P}}\right)^2 = \frac{1}{2} + \sum_{i,j=1}^{P}\frac{w_i w_j}{2P}\Omega^z_{ij} - \sum_{i=1}^{P}\frac{w_i}{\sqrt{P}}\Phi_i, \tag{D.1}$$

where we introduced the *input-label* covariance $\Phi_i \equiv \mathbb{E}\,[z_i y]$.

The expectation of the SGD update over the distribution of z is thus:

$$\mathbb{E}\;\mathrm{d}w_i = -\frac{\eta}{\sqrt{P}}\mathbb{E}\,z_i\left(\sum_{j=1}^{P}\frac{w_j z_j}{\sqrt{P}} - y\right) = \frac{\eta}{\sqrt{P}}(\Phi_i - \sum_{j=1}^{P}\frac{w_j}{\sqrt{P}}\Omega^z_{ij}) \tag{D.2}$$

Importantly, both the pmse and the average update only depend on the distribution of the features through the covariance matrix of the features Ω^z and the input label covariance Ψ.

To make progress, consider the eigen-decomposition of the covariance matrix Ω^z:

$$\Omega^z_{ij} = \frac{1}{P}\sum_{\tau}\rho_\tau \Gamma_{\tau i}\Gamma_{\tau j}, \tag{D.3}$$

where ρ_τ are the eigenvalues and Γ_τ their corresponding eigenvectors.

Define the rotation of W and Φ into this eigenbasis:

$$\tilde{w}_\tau \equiv \frac{1}{\sqrt{P}}\sum_{i=1}^{P}\Gamma_{i\tau}w_i \quad \tilde{\Phi}_\tau \equiv \frac{1}{\sqrt{P}}\sum_{i=1}^{P}\Gamma_{i\tau}\Phi_i \tag{D.4}$$

In this basis, the SGD update for the different components of $\tilde{W}$ decouple. One finds a recursive equation in which each mode evolves independently from the others:

$$\mathbb{E}\;\mathrm{d}\tilde{w}_\tau = \frac{\eta}{\sqrt{P}}(\tilde{\Phi}_\tau - \frac{1}{\sqrt{P}}\sum_{\tau=1}^{P}\rho_\tau \tilde{w}_\tau) \tag{D.5}$$

Thus, the fix point $\tilde{\hat{W}}$ such that $\mathbb{E}\ \mathrm{d}\tilde{\hat{w}}_\tau = 0$, can be found explicitly:

$$\rho_\tau \tilde{\hat{w}}_\tau = \sqrt{P}\tilde{\Phi}_\tau.$$

Rotating back in the original basis one finds the asymptotic solution for W as:

$$\hat{w}_i = \sum_{\tau;\rho_\tau>0} \frac{1}{\rho_\tau}\Gamma_{i\tau}\tilde{\Phi}_\tau. \tag{D.6}$$

The asymptotic test error is thus given by:

$$\mathsf{pmse}_{t\to\infty} = \frac{1}{2}\left(1 - \sum_\tau \frac{\tilde{\Phi}_\tau^2}{\rho_\tau}\right) \tag{D.7}$$

Asymptotic classification error From the solution of Eq. (D.6) for the asymptotic solution found by linear regression, one can obtain the asymptotic classification error performed by random features as:

$$\epsilon_{ct\to\infty} = \mathbb{E}\,\Theta(-y\lambda) = \sum_\alpha \mathcal{P}_\alpha \mathop{\mathbb{E}}_\alpha \Theta(-y\lambda), \tag{D.8}$$

where $\Theta : \mathbb{R} \to \mathbb{R}$ is the Heaviside step function and we defined $\lambda \equiv \sum_{i=1}^P \hat{w}_i z_i/\sqrt{P}$. Introducing the local field λ allows to transform the high dimensional integral over the features z into a low-dimensional (in this case one dimensional) expectation over the local field. The Gaussian equivalency theorem of Goldt et al. (2020a) shows that even though the z are not Gaussian, to leading order in $1/P$, the average Eq. (16), only depends on the first two moments λ, defined as:

$$M_\alpha = \mathop{\mathbb{E}}_\alpha[\lambda] = \sum_{i=1}^P \frac{\hat{w}_i\, \mathbb{E}_\alpha[z_i]}{\sqrt{P}} \quad Q_\alpha = \mathop{\mathsf{Cov}}_\alpha[\lambda,\lambda] = \sum_{i=1}^P \frac{\hat{w}_i\hat{w}_j}{P}\mathop{\mathsf{Cov}}_\alpha(z,z) \tag{D.9}$$

These moments can be computed analytically from the statistics of the features computed in Sec. C and from the optimal weights W obtained in Eq. (D.6). The classification error, Eq. (D.8), can thus be evaluated by means of a one dimensional integral over the distribution of λ.

$$\epsilon_{ct\to\infty} = \sum_\alpha \mathcal{P}_\alpha \int \frac{d\lambda}{\sqrt{2\pi Q_\alpha}}\Theta(-y_\alpha\lambda)e^{-\frac{1}{2Q_\alpha}(\lambda - M_\alpha)^2} \tag{D.10}$$

$$= \frac{1}{2}\left(1 - \sum_\alpha \mathcal{P}_\alpha y_\alpha \mathrm{erf}\left(\frac{M_\alpha}{\sqrt{2Q_\alpha}}\right)\right) \tag{D.11}$$

E. The three-cluster model

Similar to the analysis of the XOR-like mixture of Fig. 6, we analyse a data model with three clusters that was the subject of several recent works (Deng et al., 2019; Mai & Liao, 2019; Lelarge & Miolane, 2019; Mignacco et al., 2020a;b). The Gaussian mixture in input space can be seen in the first column of Fig. 10. The means of both positive clusters are set to 0 while the means of the negative sign clusters have first component $\pm\mu_0$ and all other $D-1$ components 0. The mixture after random feature transformation is displayed in the second column and the third and fourth column show the performance of a 2LNN, respectively, a random feature network, trained via online SGD, on this problem. Here again, we build on the observation that overparametrisation does not impact performances and train a $K = 10$ 2LNN in order to increase the number of runs that converged. The three rows, are as before, three different snr regimes, they are in order the **low**, **high** and **mixed** snr regime.

The phenomenology observed in the XOR-like mixture carries through here. In the low snr regime, $\mu_0 = \sqrt{D}$ (**top row**), the 2LNN can learn the problem and its performance remains constant with increasing input dimension. On the other hand, in this regime, the transformation performed by the random features is only linear in the large D limit. Consequently, the RF performances degrade with increasing D and are as bad as random chance in the limit of infinite input dimension. In the

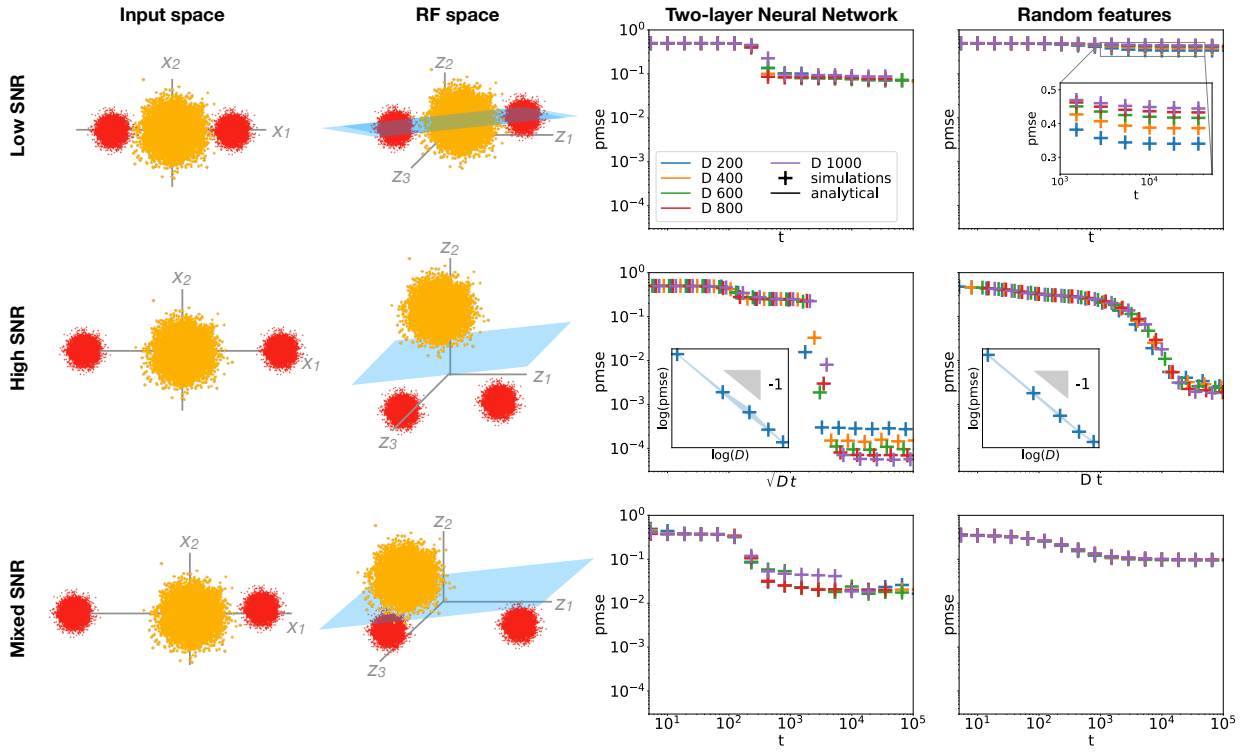

Figure 10. We compare the performance of 2LNN with $K = 10$ hidden units and ReLU activation function (**third** column) to the ones of random features (**fourth** column) on the three cluster problem with different signal-to-noise ratios. The **right** sketches the input space distribution and the **second** column the transformed distribution in feature space. 2LNN can considerably outperform RF in all three regimes. In the *low* snr regime (**top**), the action of the random features is essentially linear inducing their performance to be as bad as random guessing in the $D \to \infty$ limit. In the *high* snr regime (**middle**), instead, both networks manage to learn the task. In the *mixed* snr regime (**bottom**), the distance between opposite sign clusters remains of orther one in feature space inducing the RF performances to be unchanged with D. We plot the test error as a function of time. *Parameters:* $D = 1000$, $\eta = 0.1$, $\sigma^2 = 0.05$, $P = 2D$ for random features, $K = 10$ for 2LNN.

high snr regime instead (**second row**), where $\mu_0 = D$, the mixtures becomes well separated in feature space allowing RF to perform well. Both the performance of 2LNN and RF improve as the clusters in the mixture become more separated. The *mixed* snr regime (**bottom row**) is obtained by setting one of the negative sign clusters a distance $\sqrt{D}$ from the origin while maintaining the other one at a distance 1. Here, the random feature perform a non trivial transformation of the far away cluster while its action on the nearby cluster is linear. Hence, in feature space, one of the negative clusters remains close to the positive clusters while the other is well separated. The RF thus achieve a test error which is better than random but still worse that that of the 2LNN. Its error is constant with increasing D since it is dominated by the "spill-over" of the negative cluster into the positive cluster at the origin.

Lastly, let us comment that in all our work, we did not add a bias to the model. Adding a bias, does not change the conclusion that small 2LNN considerably outperform RF. In fact, the learning curves are only slightly modified. This is due to to our minimisation of the pmse when training the network, which, unlike classification loss that only cares about the *sign* of the estimate, penalises large differences between label $y = \pm 1$ and the output. For simplicity, we thus chose to remove the bias in our analysis, although including it is a straight forward operation.