Unanswer questions solve

一、 在 lemma 之中，都是使用 tanh 來逼近。如果使用其他 activation function，淺層網路是否能一樣逼近多項式?

主要參考文獻

1. Leshno, Lin, Pinkus, & Schocken (1993):
    - 標題：「Multilayer Feedforward Networks with a Non-Polynomial Activation Function Can Approximate Any Function」
    - 核心內容：這篇經典工作證明，只要 activation function σ不是多項式（non-polynomial），那麼單隱藏層（single hidden layer）的前饋網路就可以任意精度逼近任何連續函數。
    - 意義：這正回答你問「是不是只能用 tanh？」的問題 — 不，只要不是 polynomial activation，就有普適逼近能力。
2. Pinkus (1999):
    - 標題：「Approximation theory of the MLP model in neural networks」, Acta Numerica.
    - 核心內容：這是個 survey（回顧）文章，系統整理了 MLP（多層前饋網路）在逼近理論（approximation theory）的各種結果。特別提到 activation 函數的密度（density）問題，也就是哪些激活函數可以使網路具有 universal approximation 能力。
    - 意義：提供了理論框架和許多不同 activation 函數（不僅 tanh）的分析。
3. Sonoda & Murata (2015):
    - 標題：「Neural Network with Unbounded Activation Functions is Universal Approximator」
    - 核心內容：這篇探討了 unbounded activation functions（例如 ReLU）是否具有普適逼近能力。他們用 ridgelet 變換等工具證明了 ReLU 等激活函數仍然可以使淺層網路成為 universal

approximator。

- 意義：這是非常直接回應你關心的「換成 ReLU 可不可以」——答案是可以。

4. Neufeld & Schmocker (2024):
   - 標題：「Universal approximation results for neural networks with non-polynomial activation function over non-compact domains」
   - 核心內容：推廣了普適逼近定理到非緊緻（non-compact）領域（例如整個 $\mathbb{R}^n$ 或 L^p 空間），條件仍是激活函數是非多項式（non-polynomial）。
   - 意義：即使是在更嚴格/更廣泛的函數空間裡，非多項式 activation 函數仍然可行。
5. Zhang, Lu & Zhao (2023):
   - 標題：「Deep Network Approximation: Beyond ReLU to Diverse Activation Functions」
   - 核心內容：雖然是討論深層網路（deep），但他們證明許多常見 activation 函數（如 sigmoid, tanh, ELU, GELU, Swish …）之間可以互相轉換逼近，並且大部分 ReLU 的逼近結果可以延伸到其他 activation。
   - 意義：說明理論上不同 activation 函數之間的逼近能力是非常通用（general）的。

解釋與結論

- 激活函數不是多項式 是核心條件：根據 Leshno 等（1993）的經典理論，只要 activation function 不是多項式，淺層神經網路（single hidden layer）就能逼近任意連續函數，包括多項式。
- ReLU 也可以：儘管 ReLU 是 unbounded 且分段線性（piecewise linear），但 Sonoda & Murata (2015) 證明了它也具有普適逼近能力。

- 理論很廣：隨著後續研究（如 Neufeld & Schmocker, Zhang 等），這些逼近性質已被證實適用於很多不同 activation 函數和更廣泛的函數空間（不僅是 compact domain）。

回到你的 “lemma 作業”：

- 你在 lemma 中看到用 tanh 做逼近，這是合理且典型的一種選擇 — tanh 是非多項式，所以有理論保證可逼近多項式。
- 但 不只是 tanh 可以用：你完全可以把 activation 換成 ReLU、sigmoid、softplus、Swish 等 — 理論上淺層網路依然可以逼近你說的多項式。

連結資訊:

M. Leshno, V. Lin, S. Pinkus, S. Schocken. *“Multilayer feedforward networks with a non-polynomial activation function can approximate any function”*.
1993. https://archive.nyu.edu/bitstream/2451/14384/1/IS-91-26.pdf

S. Sonoda & N. Murata. *“Neural network with unbounded activation functions is universal approximator.”* 2017.
https://arxiv.org/abs/1505.03654

補充說明性來源:
https://en.wikipedia.org/wiki/Universal_approximation_theorem

二、 有沒有一個特別有效的學習模型，能夠去逼近非線性的資料

有效逼近非線性資料的模型與相關研究

1. Multivariate Adaptive Regression Splines (MARS)
    - Friedman, J. H. (1991). *Multivariate adaptive regression splines.* Annals of Statistics, 19(1), 1–67. DOI / 引用見：Science & Education Publishing 的摘要。
    - 介紹／解釋 MARS：Puneet Bansal & Jackson Salling (2013) "Multivariate Adaptive Regression Splines (MARS)"（課堂簡報）
    - 在實務中的應用：例如預測車輛排放中的非線性關係。
2. Kernel Regression / Kernel 方法
    - Kernel 回歸 (kernel regression)：非參數技術，用來估計非線性函數。Wikipedia 有說明 Nadaraya–Watson 估計式。
    - Kernel adaptive filter：在線 (online) 訊號處理中使用 kernel 方法來學習非線性轉換。
    - 最近模型：Kernel Stochastic Configuration Networks (KSCN) 用於非線性迴歸，具有 universal approximation 性質。
3. Contextual Regression
    - Liu, Chengyu & Wang, Wei (2017). *Contextual Regression: An Accurate and Conveniently Interpretable Nonlinear Model for Mining Discovery from Scientific Data.* arXiv. 在非線性資料上同時保有解釋能力 (interpretable) 和預測能力。
4. 非線性 SVM (Support Vector Machine)
    - Maggioni, Francesca & Spinelli, Andrea (2023). *A Robust Optimization Model for Nonlinear Support Vector Machine.* 這篇使用了 kernel SVM 來處理含不確定性的資料。 arXiv
    - Relevance Vector Machine (RVM)：類似 SVM 的 kernel 方法，

但可以產生稀疏的核表示。Bishop & Tipping（2013）的變分版本。

小結與建議

- MARS：適合高維但結構複雜、交互性強、非線性但可分段建模的資料。
- Kernel 方法（包括 kernel 回歸、kernel SVM）：理論上非常強大（可以在高維 RKHS 空間擬合複雜函數），但計算上可能較重。
- Contextual Regression：如果你除了預測能力，也希望模型有一定的可解釋性（interpretability），這是一條不錯的路。
- SVM / RVM：經典但成熟的非線性模型選項。

連結:

Contextual Regression

“Contextual Regression: An Accurate and Conveniently Interpretable Nonlinear Model for Mining Discovery from Scientific Data” — Liu, Chengyu & Wang, Wei (2017) https://arxiv.org/abs/1710.10728

MARS (Multivariate Adaptive Regression Splines)

https://en.wikipedia.org/wiki/Multivariate_adaptive_regression_spline

Friedman, J. H. (1991) 原始論文簡介 (Science & Education Publishing) https://www.sciepub.com/reference/196832

Kernel Regression

https://en.wikipedia.org/wiki/Kernel_regression

三、 如何確認訓練模型足夠精確

使用驗證集 (Validation Set)／測試集 (Test Set)

- 在訓練過程中，要保留一部分資料做驗證（validation）或測試（hold-out 测试集)；這樣可以了解模型對未見資料（unseen data）的預測能力。
- 根據 Lin Chin 的深度學習理論與實務教學，在訓練過程中須注意「訓練組－驗證組－測試組」的切分，以避免過度擬合。

交叉驗證 (Cross-Validation)

- 用 k-fold cross-validation（例如 5-fold、10-fold）反覆訓練並驗證模型，以評估模型在不同切分下的穩定性與泛化誤差。
- 留一交叉驗證（Leave-One-Out CV）是 k-fold 的極端情況，適合資料量小時使用。
- 交叉驗證在調整超參數（hyperparameter tuning）時非常有用，可以幫助你選出在驗證資料上表現最穩定／最好的模型。

學習曲線 (Learning Curves)

- 繪製訓練誤差（training error）和驗證誤差（validation error）隨訓練進行（epoch）的變化。如果訓練誤差持續下降但驗證誤差開始上升，可能是過度擬合（overfitting)。
- 如果訓練誤差和驗證誤差都很高但接近，可能是欠擬合（underfitting)，表示模型容量（capacity）太小或特徵不夠。

使用適當的評估指標 (Evaluation Metrics)

- 根據任務（分類、回歸）選擇正確的指標：分類問題常用 accuracy、F1-score、ROC-AUC；回歸問題則可能用 MSE、R^2 等。
- 若資料不平衡（class imbalance)，僅看 accuracy 可能不夠，應該看 precision, recall, F1 或 AUC。
- 針對 AUC，研究指出傳統交叉驗證可能有偏差（bias)，可以使用更穩健的方法，例如 “Tournament Leave-pair-out Cross-validation (TLPO)” 來估計 AUC。

檢查模型的可泛化性（Generalizability）

- 有研究指出，常見錯誤包括訓練／驗證／測試集之間違反獨立性（independence assumption）、選錯評估指標、批次效應（batch effects）等，這些都會導致過於樂觀（optimistic）的性能評估。
- 因此，要特別注意資料切分方式（split）、評估指標的選擇，以及是否真正模擬「未來未見資料（out-of-sample）」。

變異與穩定性（Stability）

- 多次重複交叉驗證（Repeated k-fold CV）可以觀察模型性能變異（variance），確認你的模型在不同資料子集上的表現是否穩定。
- 你也可以做早停（early stopping）：當驗證誤差在多個 epoch 上不再下降，就停止訓練，以避免過度擬合。

創新／更嚴格的驗證方法

- 有研究提到「Mutation Validation（MV）」：將訓練資料的標籤（labels）做變異（mutate），重新訓練模型，檢查模型是否對雜訊（noise）過度擬合。
- 這種方法不需傳統驗證集，也能幫助衡量模型是否「只是記住訓練資料」。

理論依據與文獻

1. **VC 維（VC Dimension）和 VC 泛化邊界**
    - 在統計學習理論中，VC 維是衡量假設空間（hypothesis space）複雜度的一個重要概念。根據 VC 理論，可導出泛化誤差上界（generalization bound）。
    - 門檻不等式（VC bound）的形式（用訓練誤差 + 模型複雜度項來界定測試誤差）：在深度／機器學習教學中也常被引用。
    - 《Statistical Learning Theory》中也講到 VC 維與樣本複雜度（sample complexity）的關係。
2. **結構風險最小化（Structural Risk Minimization, SRM）**

- SRM 是 Vapnik–Chervonenkis (VC) 理論中一個很重要的原則：透過平衡訓練誤差和模型複雜度來避免過擬合。
- 在實務上，正則化 (regularization) 就是一種實現 SRM 的方法：加入懲罰複雜度的項 (例如權重範數)，來控制模型容量。這樣做從理論上能得到更好的泛化。

3. **穩定性 (Stability) 與泛化誤差**
   - 學習演算法的「算法穩定性 (algorithmic stability)」是另一個分析泛化性能 (generalization error) 的框架。Kutin & Niyogi (2012) 論文中提出，「訓練穩定性 (training stability)」足以導出泛化誤差界。
   - 穩定性分析提供了一種不經由 VC 維但仍能界定泛化性能的方法。

4. **互信息 (Mutual Information) 與泛化誤差界**
   - 有研究將資訊論 (information theory) 用於泛化誤差界定。例如，使用訓練資料與學習算法輸出之間的 mutual information (互信息) 來界定泛化誤差。
   - 進一步有 work 將 chaining 方法 (chaining) 與 mutual information 結合，以得到更緊 (tighter) 的泛化界。

5. **Rademacher 複雜度 (Rademacher Complexity)**
   - Rademacher 複雜度 (Rademacher complexity) 是另一種衡量假設空間容量 (capacity) 的工具，可以用來給出泛化誤差上界。Wikipedia 有對這個理論做介紹。
   - 透過這種複雜度測量，我們可以對 E_train 和 E_test (或預期風險) 之間的差異 (泛化差異)做理論估計。

6. **PAC 學習 (Probably Approximately Correct, PAC)**
   - PAC 理論是統計學習理論 (computational learning theory) 的基礎框架，用來量化一個學習算法在給定樣本數、錯誤容忍度 (ε) 和信心水準 (δ) 下能否「大致正確 (approximate) 並且

高機率（probably）學到目標函數」。

- PAC 框架為「泛化誤差很小（with high probability）」提供了理論保證。

結合回你之前問的：「如何確認訓練模型足夠精確（泛化得好）」的理論支持

- 使用驗證集／交叉驗證（cross-validation）有助於估計泛化誤差，但只是實際評估方法。上述理論（VC 維、Rademacher 複雜度、穩定性、互信息、PAC）提供的是 **泛化誤差為何存在、以及上界（bounds）的理論保證**。
- 理論告訴你：模型複雜度（capacity）太高時，泛化誤差界比較大（容易 overfit）；複雜度太低（capacity 不夠）又可能欠擬合。這支持你在訓練時做 bias-variance tradeoff。
- 正則化、早停（early stopping）、交叉驗證、模型選擇（model selection）都是基於這些理論（尤其是 VC 理論和 SRM）的實務方法，用來控制泛化誤差。

連結:

Almost-everywhere algorithmic stability 與泛化誤差（Kutin & Niyogi, 2012）https://arxiv.org/abs/1301.0579

結構風險最小化（Structural Risk Minimization, SRM）

https://www.cnel.ufl.edu/wp-content/uploads/courses/EEL6814/srm.pdf

VC 維度（VC dimension）https://en.wikipedia.org/wiki/Vapnik–Chervonenkis_dimension

Argument / Algorithmic Stability（更進階），高機率穩定性與泛化界（Feldman & Vondrak, 2019）：https://arxiv.org/abs/1902.10710