Machine Learning with PyTorch and Scikit-Learn - Detailed Study Plan

Approx. 19 Weeks, ~10 hours/week (2 hours per weekday) | Primary text + optional side resources

Week(s)	Chapters & Key Sections	Key Focus Areas / Activities	Additional Resources & Notes	Side Resources (Optional)
PHASE 0: Setup & Foundations				
1	Ch. 1: Giving Computers the Ability to Learn from Data Ch. 12 (partial): PyTorch install & environment	- Understand main ML paradigms: supervised (classification, regression), unsupervised (clustering, DR), RL - Setup Python env with conda (or miniforge) and create a clean env - Install NumPy, pandas, matplotlib, scikit-learn, PyTorch (CPU/GPU) - Clone the book's code repo	Crucial setup week; verify GPU (if available) with a tiny torch.cuda.is_available() check; keep env.yml for reproducibility	PyTorch Install Guide; Conda/Miniforge quickstart; Book's GitHub code
PHASE 1: Machine Learning with Scikit-Learn				
2	Ch. 2: Training Simple ML Algorithms for Classification	- Implement Perceptron & Adaline from scratch to build intuition - Understand gradient descent; plot loss vs epochs - Work through Iris dataset; visualize decision boundaries - Learn linear separability & perceptron convergence intuition	Aim to re-derive update rules once by hand; keep notebooks tidy (data, model, plots)	3Blue1Brown: Gradient Descent; scikit-learn Perceptron docs
3	Ch. 3: A Tour of ML Classifiers Using scikit-learn	- Compare Logistic Regression, SVMs, Decision Trees, Random Forests, KNN - Practice scikit-learn's consistent API (fit/predict/score) - Understand 'no free lunch': model choice depends on data - Build a small benchmark harness for fair comparison	Track metrics beyond accuracy (precision/recall) for imbalanced cases	CS229 notes (Logistic Regression & SVM); scikit-learn User Guide (Supervised Learning)
4	Ch. 4: Building Good Training Datasets - Data Preprocessing	- Handle missing data (drop, impute) - Encode categorical vars (ordinal, one-hot) - Feature scaling (standardization) - Feature selection (L1, SFS, RF importance) - Train/test split & data leakage awareness	Preprocessing is critical; build a reusable Pipeline for transforms + model	sklearn: preprocessing & impute guides; 'Data leakage' checklists
5	Ch. 5: Dimensionality Reduction	- PCA for unsupervised compression; explained variance plots - LDA for supervised DR vs PCA - t-SNE/UMAP for visualization (don't use for downstream modeling) - Wine dataset examples	Relate PCA to eigen-decomposition & SVD; document when DR helps	3Blue1Brown: PCA; Notes on LDA; sklearn decomposition docs
6	Ch. 6: Model Evaluation & Hyperparameter Tuning	- Pipelines for clean workflows - k-fold cross-validation - Learning/validation curves (bias-variance diagnostics) - Grid/randomized search for hyperparams - Confusion matrix, ROC-AUC, PR curves, MCC; class imbalance	This is a cornerstone chapter; create a reusable CV/tuning template	sklearn: model_selection guide; ROC vs PR explainer
7	Ch. 7: Ensemble Learning	- Majority voting, Bagging (Random Forests), Boosting (AdaBoost, GBM, XGBoost) - When ensembles help; variance reduction; feature importance	Compare RF vs GBM on the same task; note training time vs accuracy	XGBoost docs; 'Elements of Statistical Learning' (bagging/boosting)

8	Ch. 8: Sentiment Analysis (NLP intro)	- Text prep: tokenization, cleaning - Bag-of-Words & TF-IDF - Out-of-core learning for large text - Topic modeling with LDA	Keep a small text-prep utility; evaluate with stratified splits	Hugging Face tokenizers overview; 'The Vector Space Model' primers
9	Ch. 9: Regression Analysis	- Simple & multiple linear regression; assumptions - Ames Housing dataset; feature engineering - Robust regression (RANSAC) - Polynomial regression; RF Regressor	Visual diagnostics: residuals, heteroscedasticity checks	Statsmodels OLS tutorial; Kaggle Ames baseline notebooks
10	Ch. 10: Clustering (Unsupervised)	- k-means/k-means++; elbow & silhouette scores - Agglomerative clustering; dendrograms/heat maps - DBSCAN for density-based clusters	Clustering is exploratory; define business/physics relevance of clusters	sklearn clustering guide; DBSCAN paper summary
PHASE 2: Deep Learning with PyTorch				
11	Ch. 11: MLP from Scratch	- Implement an MLP and backprop from scratch (MNIST) - One-hot encoding for multi-class - Training loop; evaluate with accuracy and loss curves	Derive gradients once; compare SGD vs momentum/Adam qualitatively	Michael Nielsen: 'Neural Networks and Deep Learning' Ch.2; 3Blue1Brown backprop
12	Ch. 12: PyTorch Basics	- Tensors & tensor ops; GPU vs CPU - Dataset & DataLoader; batching/shuffling - torch.nn modules; linear regression, MLP for Iris - Save/load model state_dict	Write a minimal training loop template you can reuse later	PyTorch '60-Minute Blitz'; Datasets & DataLoaders; Saving & Loading Models
13	Ch. 13: Mechanics of PyTorch	- Computation graphs & autograd - nn.Sequential vs custom nn.Module - Custom layers; small real-world projects - Intro to PyTorch Lightning as higher-level API	Refactor Week 12 code into clean modules; add type hints	PyTorch autograd docs; PyTorch Lightning quickstart
14	Ch. 14: CNNs for Images	- Convolutions, pooling, channels/filters - Implement CNNs with Conv2d/MaxPool2d - Regularization via Dropout - Data augmentation; image classification projects	Track overfitting with train/val curves; try simple augmentations first	CS231n ConvNet notes; TorchVision transforms overview
15	Ch. 15: RNNs & LSTMs for Sequential Data	- RNN mechanics; vanishing/exploding gradients - LSTMs/GRUs to address long-term dependencies - Sentiment analysis (IMDb); char-level language modeling	Clip gradients; compare vanilla RNN vs LSTM performance	Colah's blog on LSTMs; TorchText quickstart
16	Ch. 16: Transformers & Attention	- Attention mechanisms; self-attention; scaled dot-product - Transformer encoder/decoder; multi-head attention; positional encodings - Pretraining & fine-tuning (BERT/GPT/BART); HF transformers - Fine-tune a BERT for sentiment classification	Keep runs reproducible (seeds) and log with a lightweight logger	Jay Alammar: The Illustrated Transformer; Hugging Face course (Intro)

17	Ch. 17: GANs & Generative Models	- Generators vs discriminators; minimax game - Implement a simple GAN; try DCGAN - Explore Wasserstein GAN (WGAN-GP)	Use Colab/GPUs for training; monitor mode collapse symptoms	DCGAN paper summary; Hands-on GAN tutorials
18	Ch. 18: Graph Neural Networks (GNNs)	- Graphs (nodes, edges), adjacency matrices - Message passing & graph convolutions - Implement a simple GNN from scratch; PyTorch Geometric on QM9	Start with small graphs; verify shapes and batching carefully	Stanford CS224W notes; PyTorch Geometric tutorials
19	Ch. 19: Reinforcement Learning (RL)	- RL basics: agent, env, rewards; exploration vs exploitation - MDPs, returns, policies, value functions - DP/MC/TD; SARSA & Q-learning (tabular GridWorld) - Deep Q-Networks (DQN) with replay (CartPole) using Gym	Focus on fundamentals; implement vanilla Q-learning before DQN	Sutton & Barto 'Reinforcement Learning'; Spinning Up; Gym(Gymnasium) docs