Tue, Sept 2nd.

EECS 182/282A.
Prof. Gireeja Ranade.
Office hours: Tue-Th 12:30-1:30 in 400 Cory.

Today

- Machine Learning v/s Deep Learning
- Optimization

Machine Learning.

Supervised.

(x, y) data.

x: input / covariate

y: output / label.

→ Regression:

y: real number

→ Classification:

y: binary

y: discrete (multiclass)

Generally: Train a specific model for specific problem.

Unsupervised.

x: data.

No labels.

→ PCA style

Where is the variation?

→ Clustering.

- K-means.

→ Density estimation.

$x_0, x_1, x_2 \dots$

What is the distribution of the data?

Deep Learning

Supervised

Regression

Classification

Localized Annotation

semantic segmentation

Unsupervised

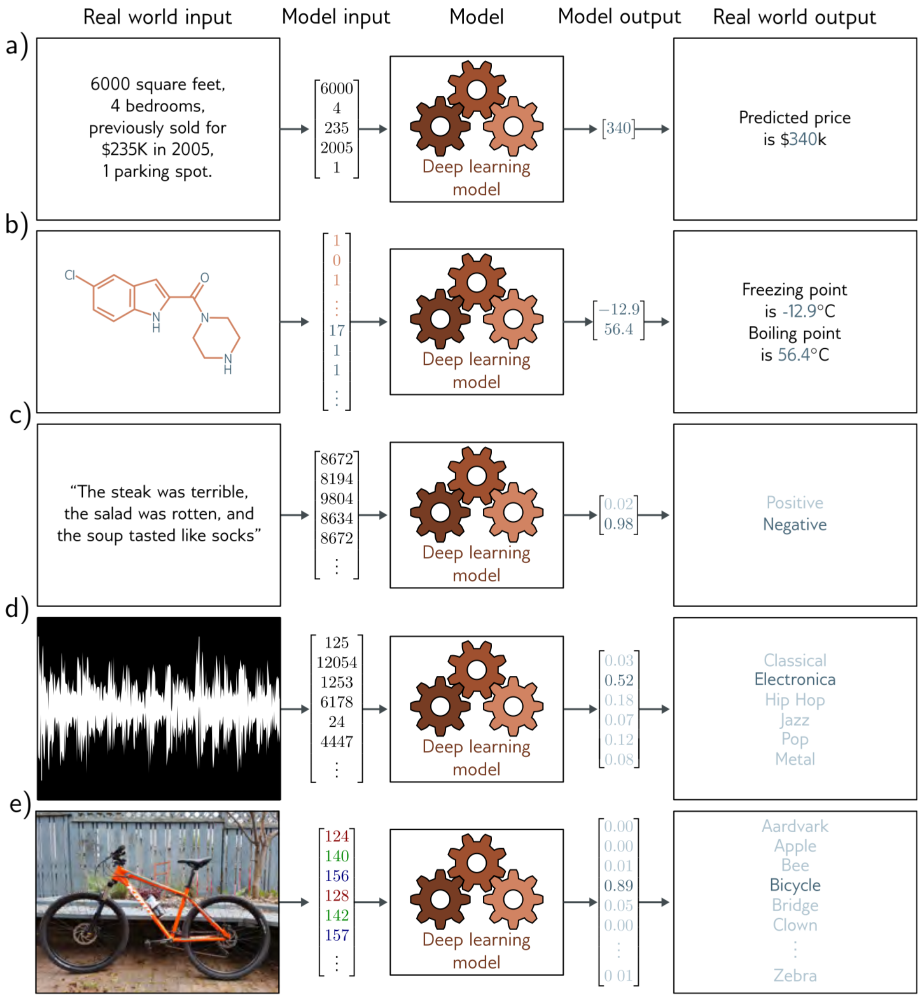
Learned embeddings

like PCA (dimensionality reduction)

Generative models

like density estimation.

Foundation models.



Sample supervised learning tasks.
Prince, 2025.

Generative models.

Figure 1.5 Generative models for images. Left: two images were generated from a model trained on pictures of cats. These are not real cats, but samples from a probability model. Right: two images generated from a model trained on images of buildings. Adapted from Karras et al. (2020b).

Optimization.

(x, y) data.

Model: $f_\theta(x) \rightarrow y$

parameters. input output.

What I have:

Training data (x_1, y_1) (x_2, y_2) ...

Empirical Risk Minimization ERM.

$$\hat{\theta} = \underset{\theta}{\operatorname{argmin}} \ \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(y_i, f_\theta(x_i))$$

Key assumption: (x, y) are drawn from $P(x, y)$

$$\theta^* = \underset{\theta}{\operatorname{argmin}} \; \mathbb{E}\left[\mathcal{L}(y_i, f_\theta(x_i)) \right]$$

Challenge 1: Don't know $P(x, y)$

Solution: Hold out test set.

Challenge 2: Loss function doesn't work with optimizers.

Challenge 3: $\hat{\theta}$ works great on training data but ~~fails~~ on test data
performs poorly

→ Regularization

Ridge regularizer: $\underset{w}{\operatorname{argmin}} \|Xw - y\|_2^2 + \lambda \|w\|_2^2$

Hyperparameter

→ New problem:

Hyperparameter search:

→ Scale hyperparameters together, couple them

→ Use what other people did.

Gradient Descent

$$\mathcal{L}_{\theta}(y, f_\theta(x)) \to \mathcal{L}(\theta, \vec{x}_{train}, \vec{y}_{train}).$$

θ_0 initial condition

$$\vec{\theta}_{t+1} = \vec{\theta}_t - \eta \cdot \nabla_\theta \mathcal{L}(\theta_t, \vec{x}_{train}, \vec{y}_{train})$$

Consider: $\mathcal{L}(\vec{w}) = \| X\vec{w} - \vec{y} \|_2^2$

Least squares: $\vec{w}^* = (X^T X)^{-1} X^T \vec{y}$

$$\nabla_{\vec{w}} \mathcal{L}(\vec{w}) = 2X^T(X\vec{w} - \vec{y})$$

$$\vec{w}_{t+1} = \vec{w}_t - 2\eta \cdot X^T(X\vec{w}_t - \vec{y}).$$

$$= (I - 2\eta X^T X)\vec{w}_t + 2\eta X^T \vec{y}$$

$$\vec{w}_{t+1} - \vec{w}_* = (I - 2\eta X^T X)\vec{w}_t + 2\eta X^T \vec{y} - \vec{w}_*$$

$$= (I - 2\eta X^T X)(\vec{w}_t - \vec{w}_*)$$

$X^T X$ is PSD.

Eigenvalues of $(I - 2\eta X^T X)$.

$|1 - 2\eta \cdot \lambda_{max}| < 1$ & $|1 - 2\eta \cdot \lambda_{min}| < 1$

$-1 < 1 - 2\eta \cdot \lambda_{max} < 1$

Similarly.

$\eta < \frac{1}{\lambda_{min}}$

$\Rightarrow -2 < -2\eta \lambda_{max}$

$\Rightarrow \eta < \frac{1}{\lambda_{max}}$

Fastest convergence: $|1 - 2\eta \cdot \lambda_{max}| = |1 - 2\eta \cdot \lambda_{min}|$.

$$\eta^* = \frac{1}{\lambda_{min} + \lambda_{max}}$$

<u>Regularization</u> $\quad \mathcal{L}(\vec{w}) = \|Xw - y\|_2^2 + \lambda \|w\|_2^2$

<u>Ridge solution:</u>

$$\vec{w}_* = (X^T X + \lambda I)^{-1} X^T \vec{y}$$

$$= X^T \underbrace{(X X^T + \lambda I)^{-1}}_{\text{Kernel ridge}} \vec{y}$$

$$X = U \Sigma V^T$$

$$\vec{w}^* = (V \Sigma^T U^T \cdot U \Sigma V^T + \lambda I)^{-1} V \Sigma^T U \cdot \vec{y}$$

$$\vdots$$

$$= \sum_{i=1}^{r} \vec{u}_i \frac{\sigma_i}{\sigma_i^2 + \lambda} \cdot \vec{u}_i^T \cdot \vec{y}$$