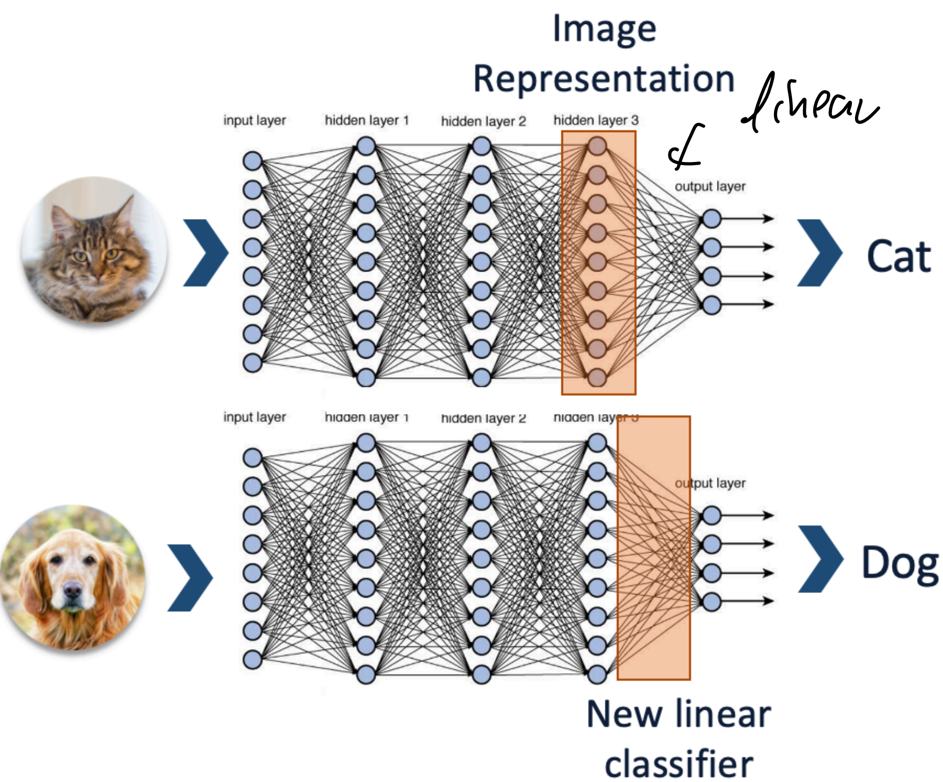
Representation Learning

Example in image representation



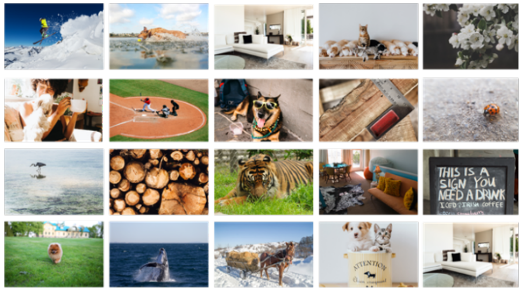
Train a neural network (ResNet) on ImageNet (1M data, 1000 classes)

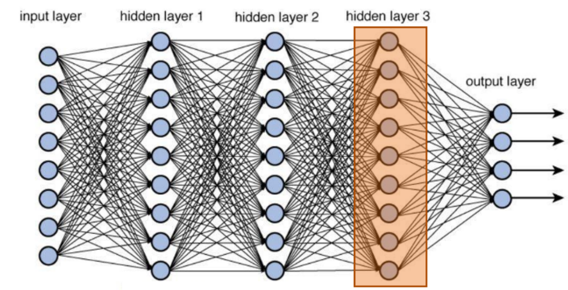
Representation (feature extractor): The mapping from image to the second-to-the-last layer.

Fix the representation, just re-train the last linear layer.

Example in image representation

Source tasks
(for training representation):
ImageNet



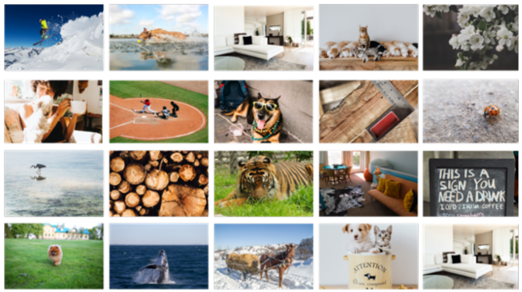


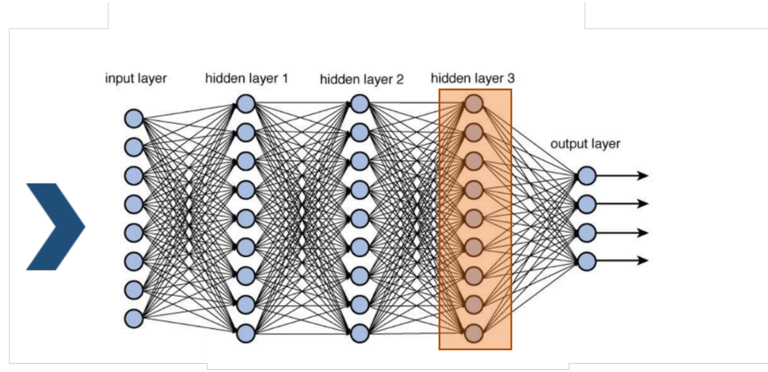
Target task:
Few-shot Learning on VOC07 dataset (20 classes, 1-8 examples per class)

- Without representation learning: **5% - 10%** (random guess = **5%**)
- With representation learning: **50% - 80%**

Example in image representation

Source tasks
(for training representation):
ImageNet





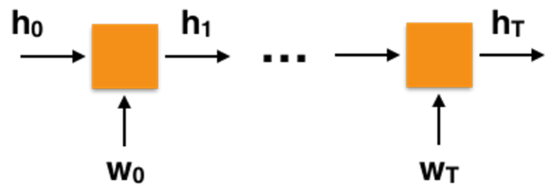
Target task:
Few-shot Learning
on VOC07 dataset
(20 classes, 1-8
examples per class)

- Without representation learning:
  5% - 10% (random guess = **5%)**
- With representation learning:
  50% - 80%

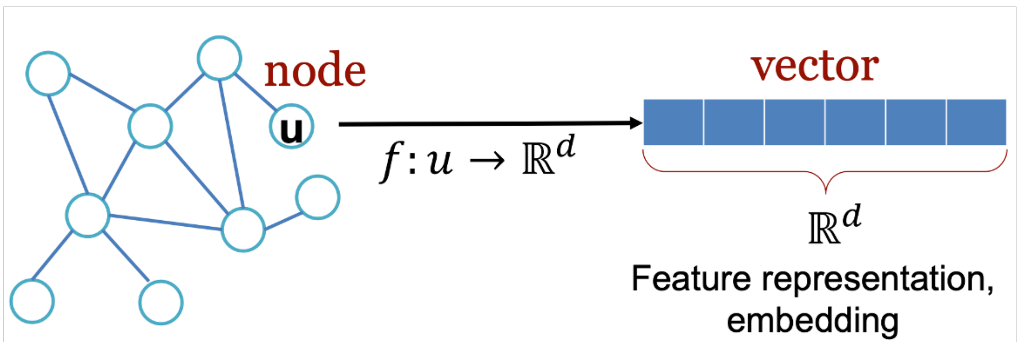
Examples

Natural Language Processing

Final hidden state:
Sentence representation



Graph Representation Learning



Representation learning

- A function that maps the raw input to a compact representation (feature vector). Learn an **embedding / feature / representation** from **labeled/unlabeled data.**
- Supervised:
  - Multi-task learning
  - Meta-learning
  - Multi-modal learning
  - …

Video = Image + audio

- Unsupervised:
  - PCA
  - ICA
  - Dictionary learning
  - Sparse coding
  - Boltzmann machine
  - Autoencoder
  - Contrastive learning
  - Self-supervised learning
  - …

$M = D \cdot A$

Desiderata for representations

Many possible answers here.

- **Downstream usability:** the learned features are "useful" for downstream tasks:
  - Example: a linear (or simple) classifier applied on the learned features only requires a small number of labeled samples. A classifier on raw inputs requires a large mount of data.

- **Interpretability:** the learned features are semantically meaningful, interpretable by a human, can be easily evaluated.
  - Not well-defined mathematically.
  - **Sparsity** is an important subcase.

Desiderata for representations

From Bengio, Courville, Vincent ’14:

- **Hierarchy / compositionality:** video/image/text are expected to have hierarchial structure: need *deep* learning.

edge detector in low layer

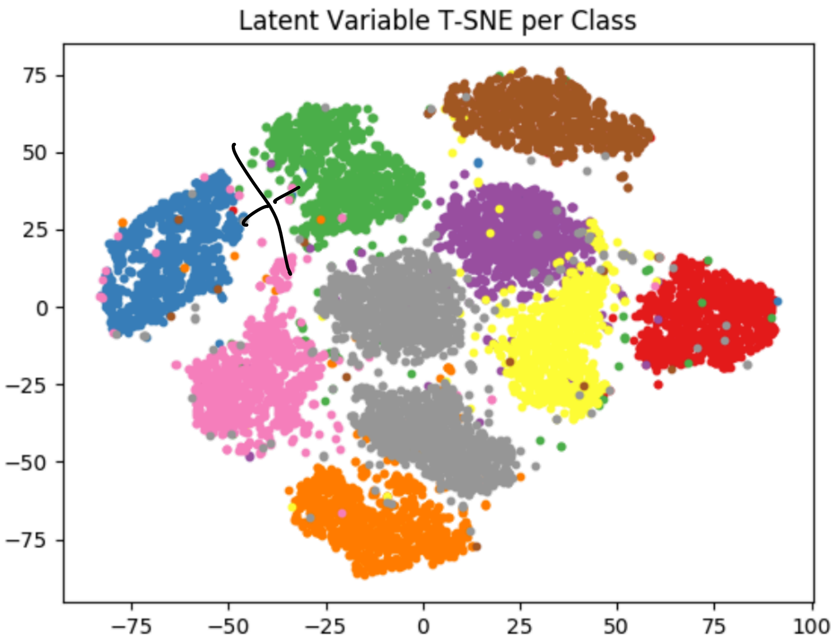
- **Semantic clusterability**: features of the same “semantic class” (e.g. images in the same class) are clustered together.

- **Linear interpolation**: in the representation space, linear interpolations produce meaningful data points (latent space is convex). Also called *manifold flattening.*

- **Disentanglement**: features capture “independent factors of variation” of data. A popular principle in modern unsupervised learning.

Semantic clustering

Semantic clusterability: features of the same "semantic class" (e.g. images in the same class) are clustered together.

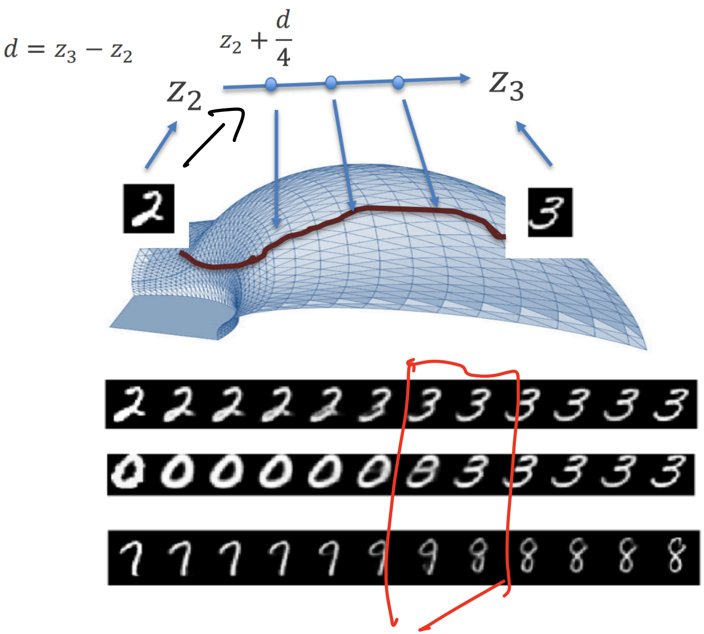


Intuition: If semantic classes are linearly separable, and labels on downstreams tasks depend linearly on semantic classes: we only need to learn a simple classifer.

t-SNE projection (a data visualization method) of VAE-learned features of 10 MNIST classes.

Linear interpolation

Linear interpolation: in the representation space, linear interpolations produce meaningful data points (latent space is convex).



Intuition: the data lies on a manifold which is complicated/ curved.

The latent variable manifold is a convex set: moving in straight lies is still on it.

Interpolations for a VAE trained feature on MNISt

Linear interpolation

Linear interpolation: in the representation space, linear interpolations produce meaningful data points (latent space is convex).

Interpolations for a BigGAN image.

Disentanglement

Disentanglement: features capture "independent factors of variation" of data (Bengio, Courville, Vincent '14).

- Very popular in modern unsupervised learning.
- Strong connections with generative models: $p_\theta(z) = \Pi_i p_\theta(z_i)$.

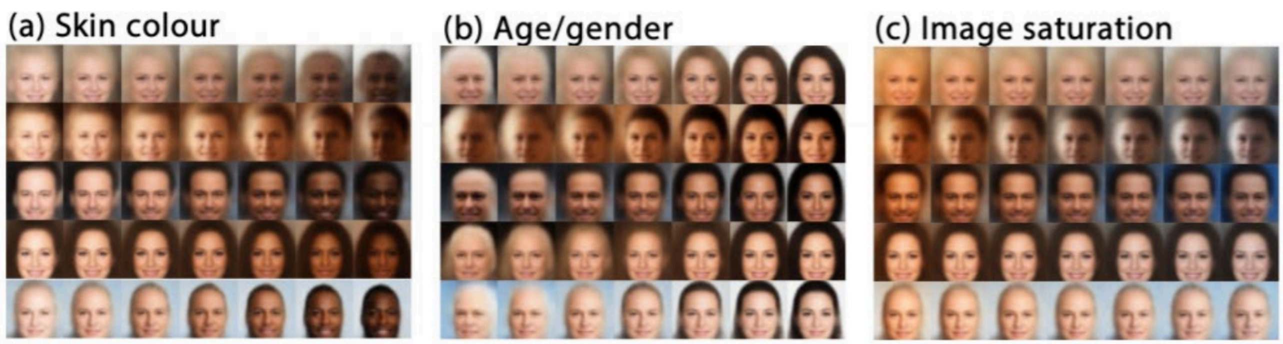
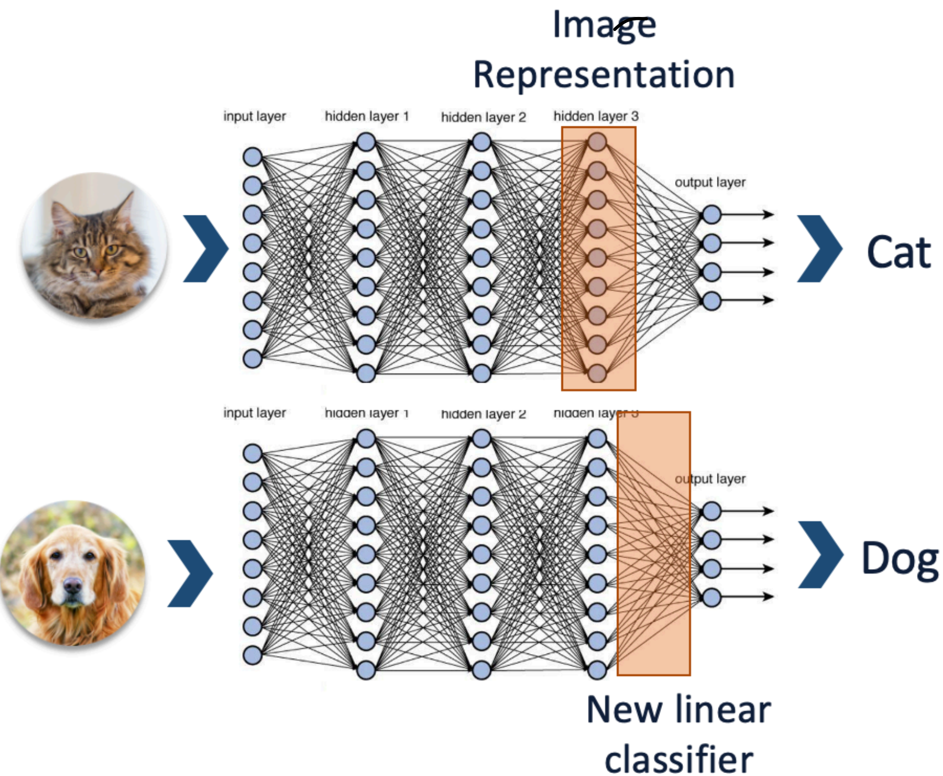


Figure 4: **Latent factors learnt by β-VAE on celebA:** traversal of individual latents demonstrates that β-VAE discovered in an unsupervised manner factors that encode skin colour, transition from an elderly male to younger female, and image saturation.

Representation Learning Methods

Multi-task representation learning

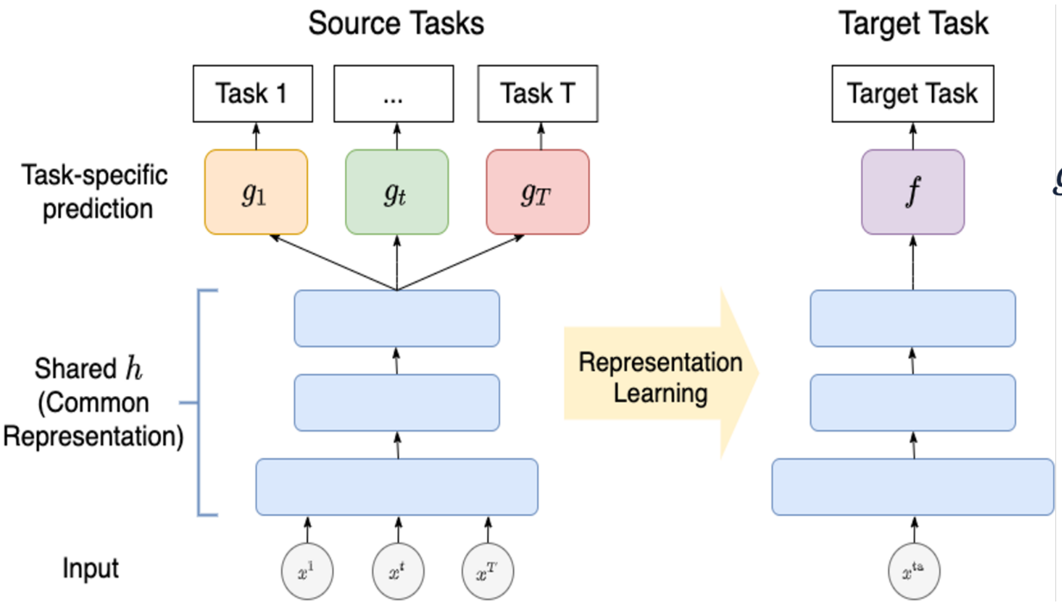


Train a neural network (ResNet) on ImageNet (1M data, 1000 classes)

Representation (feature extractor): The mapping from image to the second-to-the-last layer.

Fix the representation, just re-train the last linear layer.

Theory for multi-task representation learning



$g_1, \dots g_T, f \in \mathcal{G}$: prediction class (e.g., linear classifier)

$h \in \mathcal{H}$: representation class (e.g., multi-layer NN)

Theory for multi-task representation learning

Representation Learning

- T source tasks, each with $\boldsymbol{n_1}$ data:
  $\{(x_1^t, y_1^t) \ldots (x_{n_1}^t, y_{n_1}^t)\}_{t=1}^T$
- Learning representation:
  $$\min_{h \in \mathcal{H}} \sum_{t=1}^{T} \min_{g_t \in \mathcal{G}} \sum_{i=1}^{n_1} \ell(g_t\left(h(x_i^t)\right), y_i^t)$$

ℓ: quadratic loss

Predictor Learning

- 1 target task, with $\boldsymbol{n_2} \ll \boldsymbol{n_1}$ data:
  $(x_1^{\mathrm{ta}}, y_1^{ta}) \ldots (x_{n_2}^{ta}, y_{n_2}^{ta}) \sim \mu$
- Training for the target task:
  $$\min_{f \in \mathcal{G}} \sum_{i=1}^{n_2} \ell(f\left(h(x_i^t)\right), y_i^t)$$

Representation $\boldsymbol{h}(\cdot)$ is fixed

Review of Supervised Learning Theory

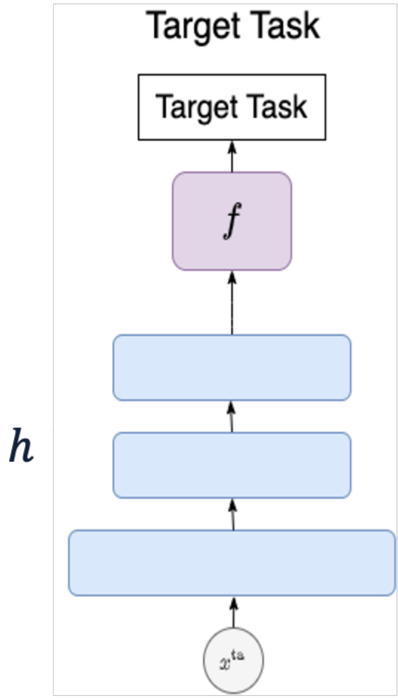
Training with data only from the target domain:

$$\min_{f\in\mathcal{G},h\in\mathcal{H}} \sum_{i=1}^{n_2} \ell\left(f\left(h\left(x_i^{ta}\right)\right), y_i^{ta}\right)$$

Theorem （Example）

$$\mathbb{E}_{(x^{ta},y^{ta})\sim\mu}\left[\ell\left(f\left(h(x^{ta})\right),y^{ta}\right)\right] = O\left(\frac{\mathcal{C}(\mathcal{H})+\mathcal{C}(\mathcal{G})}{n_2}\right)$$

$\mathcal{C}(\mathcal{H})$: complexity measure of the representation class.
$\mathcal{C}(\mathcal{G})$: complexity measure of the prediction class.
E.g., # of variables (linear function class), VC-dimension, Rademacher complexity, Gaussian width, etc



Theory for multi-task representation learning

Identify a set of (natural) assumptions:

1. If the data satisfies these assumptions, representation learning provably helps.
2. Without assumptions, representation learning does not help.

Theorem (Example)

$$\mathbb{E}_{(x^{ta},y^{ta})\sim\mu}\left[\ell\left(f\left(h(x^{ta})\right),y^{ta}\right)\right] = O\left(\frac{\mathcal{C}(\mathcal{H})}{\boldsymbol{n_1 T}} + \frac{\mathcal{C}(\mathcal{G})}{n_2}\right)$$

When **# of tasks (*T*)** is larger, much better than $O\left(\frac{\mathcal{C}(\mathcal{H}) + \mathcal{C}(\mathcal{G})}{n_2}\right)$

$\frac{\mathcal{C}(\mathcal{H})}{n_1 T}$: for learning the representation

$\frac{\mathcal{C}(\mathcal{G})}{n_2}$: for learning the predictor

Existence of a good representation

Assumption 1: Existence of a Good Representation

There exist a representation $h^* \in \mathcal{H}$ and predictors $g_1^*, g_2^*, \ldots, g_T^*, f^* \in \mathcal{G}$ such that

source $$\mathbb{E}_{(x_t, y_t) \sim \mu_t}\left[\ell\left(g_t^*\left(\boldsymbol{h}^*(x_t)\right), y_t\right)\right] = 0 \ \forall t = 1, \ldots, T$$

target $$\mathbb{E}_{(x_{ta}, y_{ta}) \sim \mu}\left[\ell\left(f^*\left(\boldsymbol{h}^*(x_{ta})\right), y_{ta}\right)\right] = 0$$

A **shared** good representation for all source tasks and the target task:

This is why we use representation learning.

(Without this assumption, we should not use representation learning)

Existence of a good representation is not enough

Source tasks:
Classify types of cats.

Target task:
Cat or dog?

Source tasks can learn a good representation for cats, but not a good representation for **both cats and dogs**.

Existence of a good representation is not enough

Input: 1000 dimensional 0/1 vector, $\{0,1\}^{1000}$

Good representation: first 100 dimension

- All tasks (source and target) only need first 100 digits for accurate prediction.
- Predicting whether the 10^{th}-digit is 1, predicting the sum of first 100 digits, etc.

Bad scenario:

- Source tasks only need to use first 50 digits: e.g., whether the 10^{th}-digit is 1
- Target tasks need to use **all** first 100 digits: e.g., predicts the sum of first 100 digits

Source tasks cannot give the **full information** about the good representation!

Theory for multi-task representation learning

$\mathcal{G}$: linear prediction class (last layer of neural networks)

Assumption 1: Existence of a Good Representation

There exist a representation $h^* \in \mathcal{H}, h^*(x) \in \mathbb{R}^k$ and $w_1^*, w_2^*, \ldots, w_T^*, w_{ta}^* \in \mathbb{R}^k$:

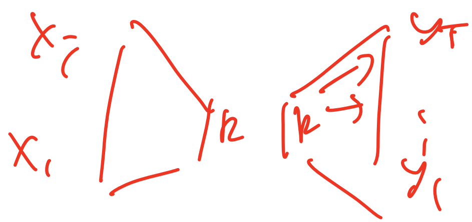
$$\mathbb{E}_{(x_t,y_t)\sim\mu_t}[\ell(\langle w_t^*, h^*(x_t)\rangle, y_t)] = 0 \;\forall t = 1, \ldots, T$$

$$\mathbb{E}_{(x_{ta},y_{ta})\sim\mu}[\ell(\langle w_{ta}^*, h^*(x_{ta})\rangle, y_{ta})] = 0$$

Assumption 2: Diversity of Source Tasks for Linear Predictor

$W^* = [w_1^*, w_2^*, \ldots, w_T^*] \in \mathbb{R}^{k\times T}$ is full rank (=k).

Need $T \geq k$: cover the span of the good representation.



Theory for multi-task representation learning

Assumption 1: Existence of a Good Representation

There exist a representation $h^* \in \mathcal{H}, h^*(x) \in \mathbb{R}^k$ and $w_1^*, w_2^*, \ldots, w_T^*, w_{ta}^* \in \mathbb{R}^k$:

$$\mathbb{E}_{(x_t, y_t)\sim\mu_t}[\ell(\langle w_t^*, h^*(x_t)\rangle, y_t)] = 0 \; \forall t = 1, \ldots, T$$

$$\mathbb{E}_{(x_{ta}, y_{ta})\sim\mu}[\ell(\langle w_{ta}^*, h^*(x_{ta})\rangle, y_{ta})] = 0$$

Theorem [D. Hu Kakade Lee Lei, 2020]

Under Assumption 1 &2, we have $\mathbb{E}_{(x^{ta}, y^{ta})\sim\mu}\left[\ell\left(f\left(h(x^{ta})\right), y^{ta}\right)\right] = O\left(\frac{\mathcal{C}(\mathcal{H})}{n_1 T} + \frac{k}{n_2}\right)$.

$\mathcal{C}(\mathcal{H})$: Gaussian width of the representation class $\mathcal{H}$.

- Measures how well the function in the class can fit the noise.