Forms of weak supervision

I Small data

Data augmentation:

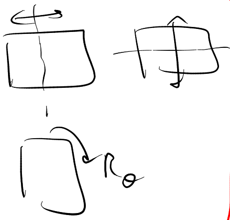
ex: image classification

add transformations

↓

label: invariant
equivariant

dataset: 

eg: - flip: 

- rotation: R_θ

- translate

- crop

1 data point ↓ 4 points → same label

geometric transforms

- color balance
- pixel noise
- contrast

} value-based transforms

Use a simulator

- generate a lot of data
- how realistic? (mind the gap)

Multi-tasking

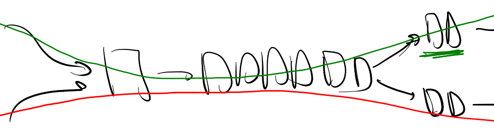
- one real task + one (several) auxiliary tasks

real task

aux. one

Few labeled data

Lots of labeled data



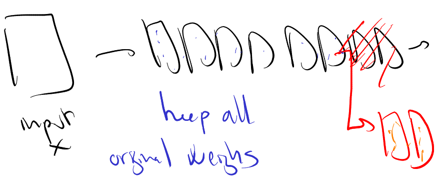
→ real task → supervised (few points)

→ aux. task → supervised with lots of data

Transfer learning

sequential training: first: on auxiliary task
then: on real task

eg: classify images
rare animals (few pictures)
↓
ImageNet: 1000 classes
× 1000 images/class
↳ lots of
= networks
already trained



input x → y aux. task

→ $\tilde{y}$ new task

keep all original weights

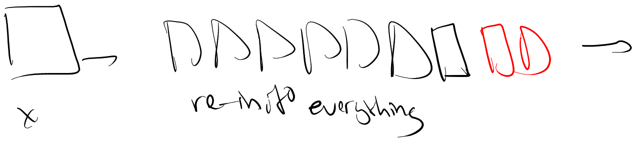
train new layers only: in matter of seconds/minutes

↓

unfreeze after convergence of new layers

(fine-tuning)

minutes / × #times trying ≠ hyper-parameters

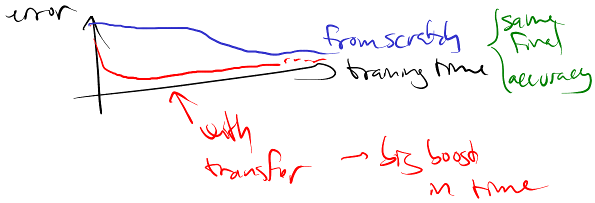


→ hours (x #times) ResNet
days VGG
months build your own

Analysis from [Rethinking ImageNet pre-training]

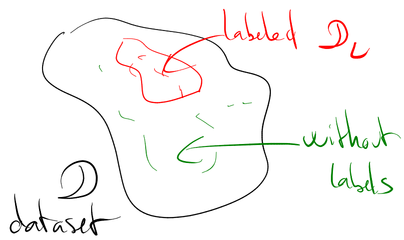
↙ small data:
helps in getting good features

↘ big data:
≠ from training from scratch



II Forms of weak supervision

amount of data, but only few are labeled



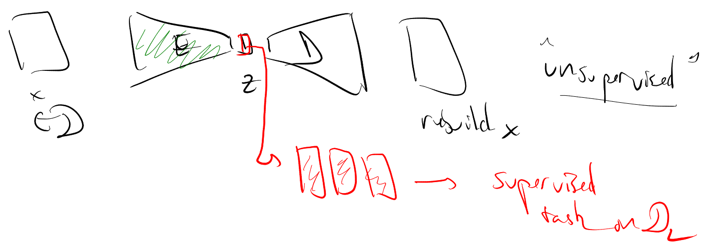
ex: when labeling is costly
(requires time, expert knowledge...)

⇒ "semi-supervised"

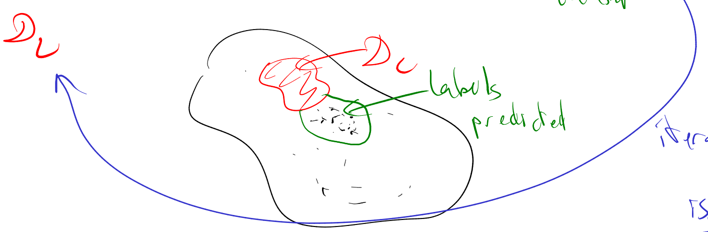
Several approaches:

1) unsupervised training on full dataset $\mathcal{D}$ → good representation → supervised task on $\mathcal{D}_L$

ex: auto-encoder:

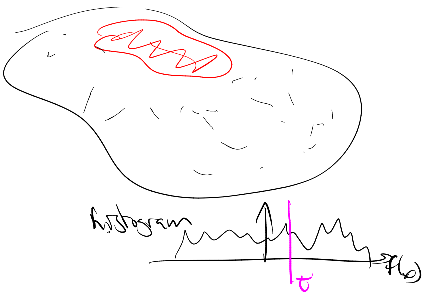


2) supervised training $\mathcal{D}_L$ → label some of the non-labeled data → bigger dataset $\mathcal{D}'_L = \mathcal{D}_L \cup \mathcal{D}_{new\ labels}$



issue: what about confident mistakes?

3) supervised training on $\mathcal{D}_L$ → apply network to full dataset → check statistics / properties of your estimator / predictor & adjust global parameters



ex: global bias also: density, margin (svm)

binary class$^{\circ}$: A / B

40% 60% target

$\mathbb{1}(f(x) > \tau)$ ← adapt threshold τ

55% 45% current estimator

Self-supervision (approach 1)

↳ used for pre-training

- no supervision ⇒ design supervised task (aux) so that labels are automatically assigned

eg: image classif°

real task → classify

puzzle: scramble 9 pieces → permuted image → dummy task: where was the middle piece? ↳ 9 classes

rotation: x → apply random rotat° → $R_\theta(x)$ → what was the angle θ? (regression task)

eg: video classif°:

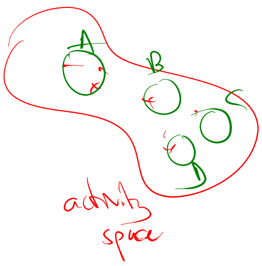
- predict next frame
- give 3 frames: ask temporal order?

Teacher-student techniques (distillation)

"ClusterFit":

x → → $\hat{y}$

↳ cluster



train new network to predict cluster class

x → → cluster index

DINO:

student: x → randomly initialized (beware architecture bias)

$x' =$ augmented x

teacher: x → average of the last past students

ex: $x' = R_\theta(x)$ or noisy x] group of transforms

ask student (x') to be close to teacher (x)

⇓

learning to be invariant to the group of transform

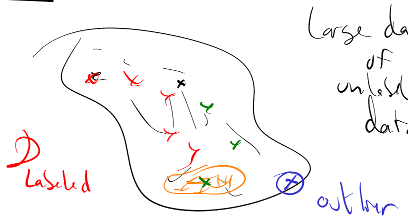
⚠ ask output to have high variance (to prevent collapse)

then:

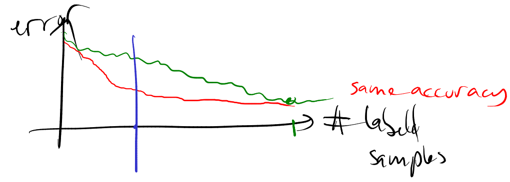
train linear classifier on top

⇒ almost as good as supervised techniques

Active learning



→ ask someone to label more samples (costly)



Goal: pick the most informative samples to label
↳ loss improvement over all dataset

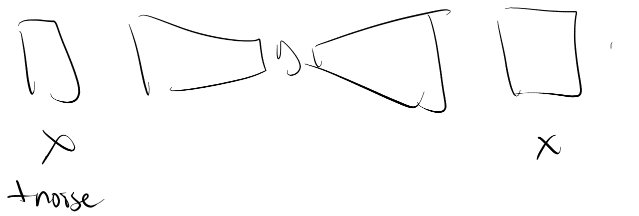
Modeling 2 quantities:

- how non-confident are current predictions for that sample
- how similar the sample is to many other unlabeled samples

IV Noisy data

1) Noisy inputs

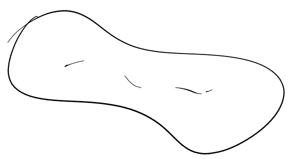
→ denoising auto-encoder



→ learn to get rid of the noise

⚠ model the noise

2) Classificat° with noisy labels



Fully labeled set

Labels are often wrong

if 1% labels wrong: ok

10%,

50%,

55% —— random: noise cancels out on average

3) Regression with noisy labels

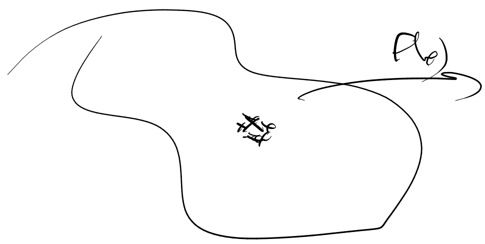
noise: centered (no bias)

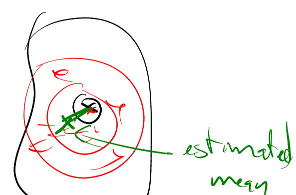
⇓

quantify similarity

⇓

show: NTK





Labels

L_2 regression

$$\sum_i \|f(x_i) - y_i\|^2$$

$\hookrightarrow f(x) = \bar{y} \pm 1/\sqrt{N}$ ← number of similar samples