~~Towards Lifelong Machine Learning~~
Multi-Task and Lifelong Learning with Unlabeled Tasks

Christoph Lampert

IST AUSTRIA
Institute of Science and Technology

HSE Computer Science Colloquium
September 6, 2016

IST Austria (Institute of Science and Technology Austria)

http://www.ist.ac.at

- institute for **basic research**
- located in outskirts of Vienna
- English-speaking

Research at IST Austria

- focus on interdisciplinarity
- currently 41 research groups:
  - Computer Science, Mathematics,
  - Physics, Biology, Neuroscience

Open Positions

- Faculty (Asst.Prof/Prof)
- Postdocs
- PhD (4yr Graduate School)
- Internships

My research group at IST Austria:


Alex Kolesnikov


Georg Martius


Asya Pentina


Amélie Royer


Alex Zimin

Funding Sources:

IST AUSTRIA
Institute of Science and Technology

erc

Research Topics

Theory (Statistical Machine Learning)

- Multi-task learning
- Lifelong learning (learning to learn)
- Domain adaptation
- Learning with dependent data

Models/Algorithms

- Zero-shot learning
- Weakly-supervised learning
- Classifier adaptation
- Probabilistic graphical models

Applications (in Computer Vision)

- Object recognition
- Image segmentation
- Object localization
- Semantic image representations

Research Topics

Theory (Statistical Machine Learning)

- Multi-task learning
- Lifelong learning (learning to learn)
- Domain adaptation
- Learning with dependent data

Models/Algorithms

- Zero-shot learning
- Weakly-supervised learning
- Classifier adaptation
- Probabilistic graphical models

Applications (in Computer Vision)

- Object recognition
- Image segmentation
- Object localization
- Semantic image representations

Today's colloquium

Research Topics

Theory (Statistical Machine Learning)

- Multi-task learning
- Lifelong learning (learning to learn)
- Domain adaptation
- Learning with dependent data

Models/Algorithms

- Zero-shot learning
- Weakly-supervised learning
- Classifier adaptation
- Probabilistic graphical models

Applications (in Computer Vision)

- Object recognition
- Image segmentation
- Object localization
- Semantic image representations

Today's colloquium

Mon/Wed/Fri 16:40 HSE

Research Topics

Theory (Statistical Machine Learning)

- Multi-task learning
- Lifelong learning (learning to learn)
- Domain adaptation
- Learning with dependent data

Models/Algorithms

- Zero-shot learning
- Weakly-supervised learning
- Classifier adaptation
- Probabilistic graphical models

Applications (in Computer Vision)

- Object recognition
- Image segmentation
- Object localization
- Semantic image representations

Today's colloquium

Mon/Wed/Fri 16:40 HSE

Thu, 12:00, Skoltech

Research Topics

Theory (Statistical Machine Learning)

- Multi-task learning
- Lifelong learning (learning to learn)
- Domain adaptation
- Learning with dependent data

Models/Algorithms

- Zero-shot learning
- Weakly-supervised learning
- Classifier adaptation
- Probabilistic graphical models

Applications (in Computer Vision)

- Object recognition
- Image segmentation
- Object localization
- Semantic image representations

Today's colloquium

Mon/Wed/Fri 16:40 HSE

Thu, 12:00, Skoltech

Thu, 19:00, Yandex

Research Topics

Theory (Statistical Machine Learning)

- Multi-task learning
- Lifelong learning (learning to learn)
- Domain adaptation
- Learning with dependent data

Models/Algorithms

- Zero-shot learning
- Weakly-supervised learning
- Classifier adaptation
- Probabilistic graphical models

Applications (in Computer Vision)

- Object recognition
- Image segmentation
- Object localization
- Semantic image representations

Today's colloquium

Mon/Wed/Fri 16:40 HSE

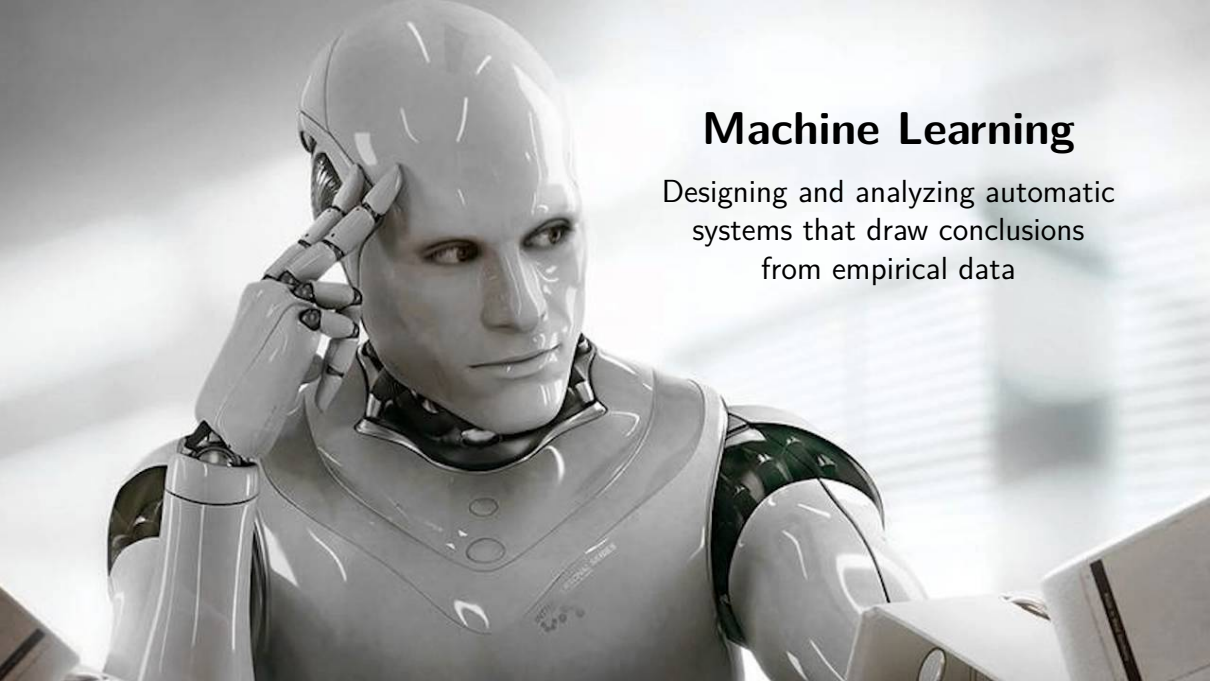
Thu, 12:00, Skoltech

Thu, 19:00, Yandex

Fri, 14:00, MSU

Overview

- Introduction
- Learning Multiple Tasks
- Multi-Task Learning with Unlabeled Tasks

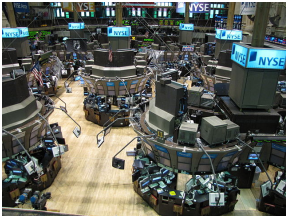


Machine Learning

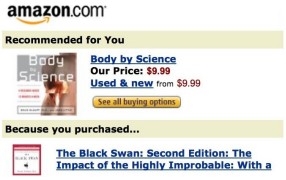
Designing and analyzing automatic systems that draw conclusions from empirical data

What is "Machine Learning"?

Robotics



Time Series Prediction



Social Networks

Language Processing

Healthcare

Natural Sciences

Images: NASA, Kevin Hutchinson (CC BY-SA 2.0), Amazon, Microsoft, Ragesoss (CC BY-SA 3.0), R. King

What is "Machine Learning"?

Image: "The CSIRO GPU cluster at the data centre" by CSIRO. Licensed under CC BY 3.0 via Wikimedia Commons

What is "Machine Learning"?

A. M. Turing (1950) Computing Machinery and Intelligence. *Mind 49:* 433-460.

COMPUTING MACHINERY AND INTELLIGENCE

By A. M. Turing

1. The Imitation Game

I propose to consider the question, "Can machines think?" This should begin with definitions of the meaning of the terms "machine" and "think." The definitions might be framed so as to reflect so far as possible the normal use of the words, but this attitude is dangerous, If the meaning of the words "machine" and "think" are to be found by examining how they are commonly used it is difficult to escape the conclusion that the meaning and the answer to the question, "Can machines think?" is to be sought in a statistical survey such as a Gallup poll. But this is absurd. Instead of attempting such a definition I shall replace the question by another, which is closely related to it and is expressed in relatively unambiguous words.

Introduction/Notation

Learning Tasks

A **learning tasks**, $\mathcal{T}$, consists of

- **input set**, $\mathcal{X}$ (e.g. email text, or camera images),
- **output set**, $\mathcal{Y}$ (e.g. $\{$`spam`, `not spam`$\}$ or $\{$`left`, `right`, `straight`$\}$)
- **loss function**, $\ell(y, \bar{y})$ (*"how bad is it to predict $\bar{y}$ if y is correct?"*),
- **probability distribution**, $D(x, y)$, over $\mathcal{X} \times \mathcal{Y}$.

We assume that $\mathcal{X}$, $\mathcal{Y}$ and ℓ are know, but $D(x, y)$ is unknown.

Learning Tasks

A **learning tasks**, $\mathcal{T}$, consists of

- **input set**, $\mathcal{X}$ (e.g. email text, or camera images),
- **output set**, $\mathcal{Y}$ (e.g. $\{$`spam`, `not spam`$\}$ or $\{$`left`, `right`, `straight`$\}$)
- **loss function**, $\ell(y, \bar{y})$ (*"how bad is it to predict $\bar{y}$ if y is correct?"*),
- **probability distribution**, $D(x, y)$, over $\mathcal{X} \times \mathcal{Y}$.

We assume that $\mathcal{X}$, $\mathcal{Y}$ and ℓ are know, but $D(x, y)$ is unknown.

Learning

The goal of **learning** is to find a function $h : \mathcal{X} \to \mathcal{Y}$ with minimal **risk** (expected loss)

$$\text{er}(h) := \mathbb{E}_{(x,y)\sim D}\ \ell(\, y, h(x)\,)$$

(also called: "generalization error")

Supervised Learning

Given a **training set**

$$S = \{(x_1, y_1), \ldots, (x_n, y_n)\}$$

consisting of independent samples from D, i.e. $S \overset{i.i.d.}{\sim} D$, define the **empirical risk**

$$\widehat{\mathsf{er}}(h) := \frac{1}{n} \sum_{i=1}^{n} \ell(\, y_i, h(x_i)\,)$$

(also called: "training error")

Empirical Risk Minimization

Let $\mathcal{H} \subset \{h : \mathcal{X} \to \mathcal{Y}\}$ be a **hypothesis set**. Finding a hypothesis $h \in \mathcal{H}$ by solving

$$\min_{h \in \mathcal{H}} \widehat{\mathsf{er}}(h)$$

is called **empirical risk minimization (ERM)**.

Empirical Risk Minimization

Let $\mathcal{H} \subset \{h : \mathcal{X} \to \mathcal{Y}\}$ be a **hypothesis set**. Finding a hypothesis $h \in \mathcal{H}$ by solving

$$\min_{h \in \mathcal{H}} \widehat{\mathrm{er}}(h)$$

is called **empirical risk minimization (ERM)**.

Fundamental Theorem of Statistical Learning Theory

[*Alexey Chervonenkis* and *Vladimir Vapnik*, 1970s]

"ERM is a good learning strategy if
and only if $\mathcal{H}$ has finite VC-dimension."

Vapnik-Chervonenkis generalization bound

Let $\mathcal{H}$ be a set of binary-valued hypotheses. For any $\delta > 0$, the following inequality holds with probability at least $1 - \delta$ over the draw of a training set

$$|\mathsf{er}(h) - \widehat{\mathsf{er}}(h)| \leq \sqrt{\frac{\mathrm{VC}(\mathcal{H})\left(\ln \frac{2n}{\mathrm{VC}(\mathcal{H})} + 1\right) + \log(4/\delta)}{n}}$$

simultaneously for all $h \in \mathcal{H}$.

Vapnik-Chervonenkis generalization bound

Let $\mathcal{H}$ be a set of binary-valued hypotheses. For any $\delta > 0$, the following inequality holds with probability at least $1 - \delta$ over the draw of a training set

$$\mathrm{er}(h) \leq \widehat{\mathrm{er}}(h) + \sqrt{\frac{\mathrm{VC}(\mathcal{H})\left(\ln \frac{2n}{\mathrm{VC}(\mathcal{H})} + 1\right) + \log(4/\delta)}{n}}$$

simultaneously for all $h \in \mathcal{H}$.

Observation (assuming our training set is not atypical):

- the right hand side of the bound contains **computable quantities**.
  $\rightarrow$ we can compute the hypothesis that minimizes it
- the bound holds **uniformly** for all $h \in \mathcal{H}$ $\quad \rightarrow$ it also holds for the minimizer

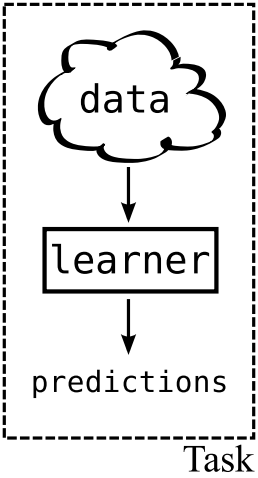
Choosing the hypothesis that minimizes $\widehat{\mathrm{er}}(h)$ is a promising strategy $\rightarrow$ ERM

Summary

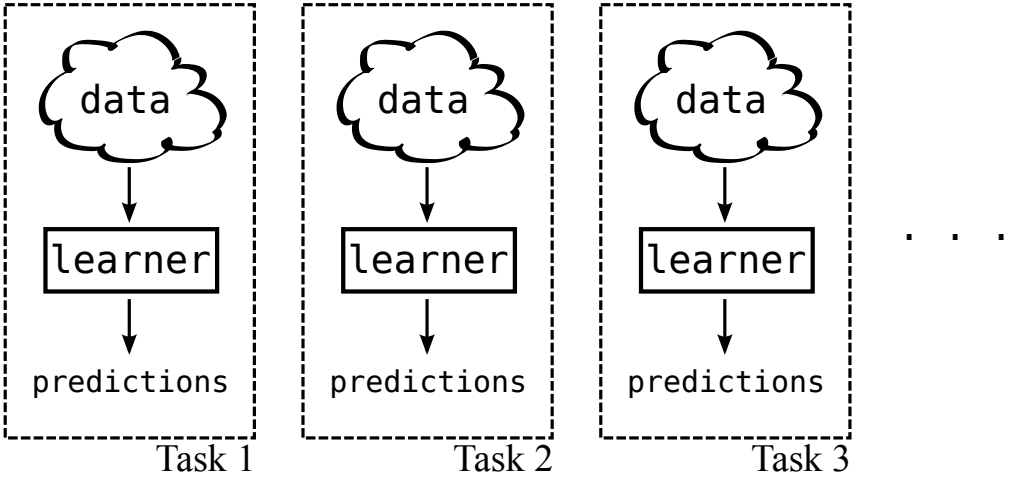
- for some learning algorithms, it is possible to give (probabilistic) guarantees of their performance on future data.
- a principled way to construct such algorithms:
  1) prove a generalization bound that
     - consists only of computable quantities on the right hand side
     - holds uniformly over all hypotheses
  2) algorithm $\equiv$ minimizing the r.h.s. of the bound with respect to hypotheses
- even when not practical, still a good way just to get *inspiration* for new algorithms

Learning multiple tasks

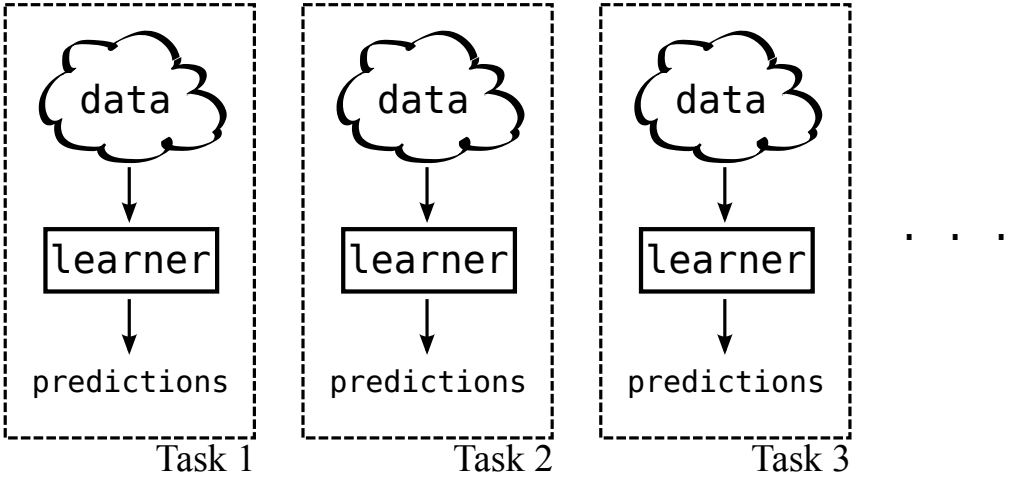
Learning a Single Task



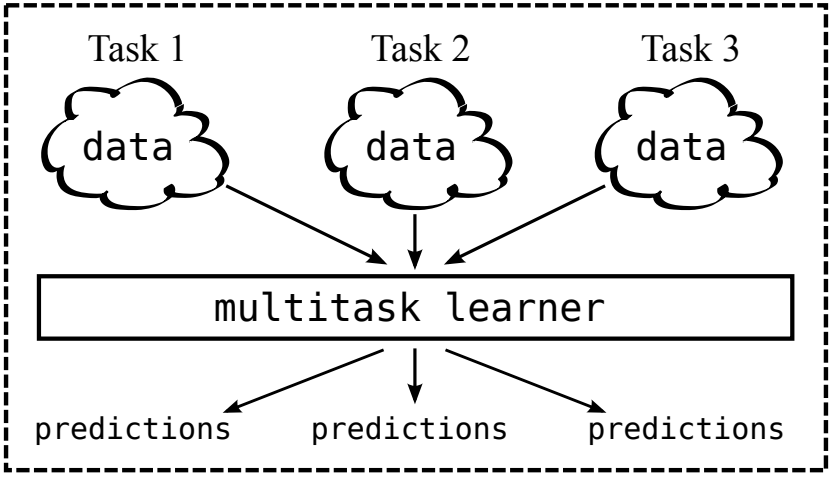
Learning Multiple Tasks



Learning Multiple Tasks

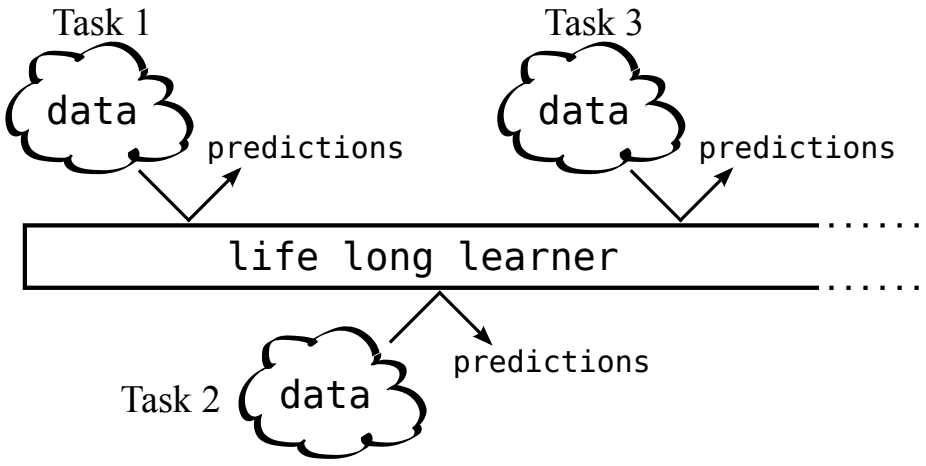


Tabula Rasa Learning



Multi-Task Learning

Future challenge: continuously improving open-ended learning



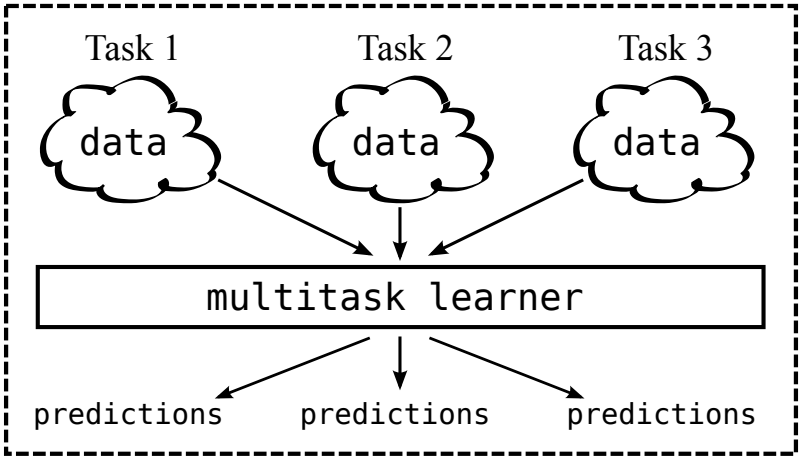
Lifelong Learning

Active Task Selection for Multi-Task Learning

Asya Pentina

A. Pentina, CHL, *"Active Task Selection for Multi-Task Learning"*, arXiv:1602.06518 [stat.ML]

Multi-Task Learning



Example: Personalized Speech Recognition

each user speaks differently: learn an individual classifier

Example: Personalized Spam Filters

each user has different preferences: learn individual filter rules

Multi-Task Learning: state-of-the-art

Sharing Parameters

- assume: all tasks are minor variations of each other
- e.g. introduce *regularizer* that encourages similar weight vectors

Sharing Representations

- assume: one set of features works well for all tasks
- e.g., neural network with early layers shared between all tasks

Sharing Samples

- form larger training sets from samples sets of different tasks
- extreme case: merge all data and learn a single classifier

All have in common: they need labeled training sets for all tasks

New Setting:

- Given: unlabeled sample sets for each of T tasks
- Algorithm chooses subset of k tasks and asks for labeled samples
- Output: classifiers for all tasks (labeled as well as unlabeled)

New Setting:

- Given: unlabeled sample sets for each of T tasks
- Algorithm chooses subset of k tasks and asks for labeled samples
- Output: classifiers for all tasks (labeled as well as unlabeled)

Example: speech recognition, $T =$ millions

- collecting some unlabeled data (=speech) from each user is easy
- asking users for annotation is what annoys them: avoid as much as possible
- strategy:
  - select a subset of speakers
  - ask only those for training annotation
  - use data to build classifiers for all
- is this going to work?
  - clearly not, if all speakers are completely different
  - clearly yes, if all speakers are the same

Can we quantify this effect, give guarantees and derive a principled algorithm?

Reminder: Learning Tasks

A learning tasks, $\mathcal{T}$, consists of

- input space, $\mathcal{X}$ (e.g. audio samples)
- output space, $\mathcal{Y}$ (e.g. phonemes or natural language text)
- loss function $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$
- joint probability distribution, $D(x, y)$, over $\mathcal{X} \times \mathcal{Y}$

For simplicity, assume

- $\mathcal{Y} = \{0, 1\}$ (binary classification tasks)
- $\ell(y, \bar{y}) = [\![y \neq \bar{y}]\!]$ ($0/1$-loss)
- "realizable" situation:
  - x distributed according to a (marginal) distribution $D(x)$
  - deterministic labeling function $f : \mathcal{X} \to \mathcal{Y}$

Learning Classifiers for All Tasks

Learning Classifiers for Labeled Tasks

For labeled tasks, we have a training set $S_j = \{(x_1, y_1), \ldots, (x_n, y_n)\}$, we can compute a classifier, e.g. by minimizing the empirical risk

$$\widehat{\mathsf{er}}(h) := \frac{1}{n} \sum_{i=1}^{n} \ell(\, y_i, h(x_i)\,).$$

Other algorithms instead of ERM: support vector machines, deep learning, ...

Learning Classifiers for Unlabeled Tasks

For unlabeled tasks, we have only data $S_j = \{x_1, \ldots, x_n\}$ without annotation, so we cannot compute the empirical risk to learn or evaluate a classifier.

Main idea: let's re-use a classifier from a 'similar' labeled tasks.

Given two tasks, $\langle D_1, f_1 \rangle$ and $\langle D_2, f_2 \rangle$. How to determine how similar they are?

A similarity measure between tasks (first attempt)

Compare $D_1(x, y)$ and $D_2(x, y)$, e.g., by their Kullback-Leibler divergence

$$\mathrm{KL}(D_1 | D_2) = \mathbb{E}_{D_1}[\log \frac{D_1}{D_2}]$$

Problem: too strict, e.g., $\mathrm{KL}(D_1 | D_2) = \infty$ unless $f_1 \neq f_2$

Given two tasks, $\langle D_1, f_1 \rangle$ and $\langle D_2, f_2 \rangle$. How to determine how similar they are?

A similarity measure between tasks (first attempt)

Compare $D_1(x, y)$ and $D_2(x, y)$, e.g., by their Kullback-Leibler divergence

$$\mathrm{KL}(D_1 | D_2) = \mathbb{E}_{D_1}[\log \frac{D_1}{D_2}]$$

Problem: too strict, e.g., $\mathrm{KL}(D_1 | D_2) = \infty$ unless $f_1 \neq f_2$

A similarity measure between tasks (second attempt)

Compare only at marginal distributions, e.g. Kullback-Leibler divergence, and treat labeling functions separately.

Problem: still very strict, can't be estimated well from sample sets

A more suitable similarity measure between tasks:

Discrepancy Distance [Mansour *et al*. 2009]

For a set of hypothesis $\mathcal{H}$ and $\mathrm{er}_D(h, h') = \mathbb{E}_{x\sim D}[\![h(x) \neq h'(x)]\!]$.

$$\mathrm{disc}(D_1, D_2) = \max_{h,h'\in\mathcal{H}} \left|\mathrm{er}_{D_1}(h, h') - \mathrm{er}_{D_2}(h, h')\right| .$$

- maximal classifier disagreement between distributions D_1, D_2
- only as strong as we need when dealing with a hypothesis class $\mathcal{H}$

A more suitable similarity measure between tasks:

Discrepancy Distance [Mansour *et al*. 2009]

For a set of hypothesis $\mathcal{H}$ and $\mathrm{er}_D(h, h') = \mathbb{E}_{x\sim D}[\![h(x) \neq h'(x)]\!]$.

$$\mathrm{disc}(D_1, D_2) = \max_{h,h'\in\mathcal{H}} \left|\mathrm{er}_{D_1}(h, h') - \mathrm{er}_{D_2}(h, h')\right| .$$

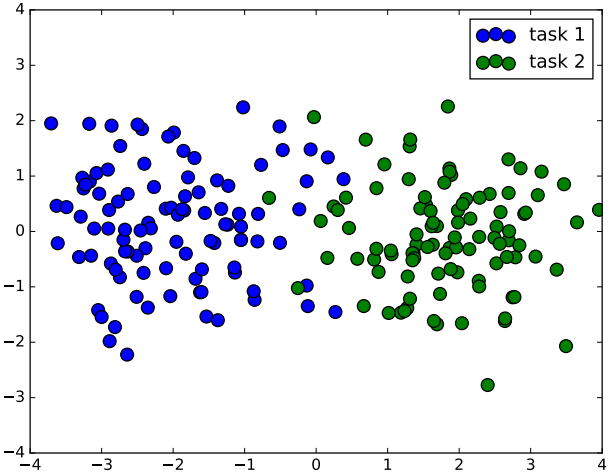
- maximal classifier disagreement between distributions D_1, D_2
- only as strong as we need when dealing with a hypothesis class $\mathcal{H}$
- can be estimated from (unlabeled) sample sets S_1, S_2:

$$\widehat{\mathrm{disc}}(S_1, S_2) = 2(1 - E)$$

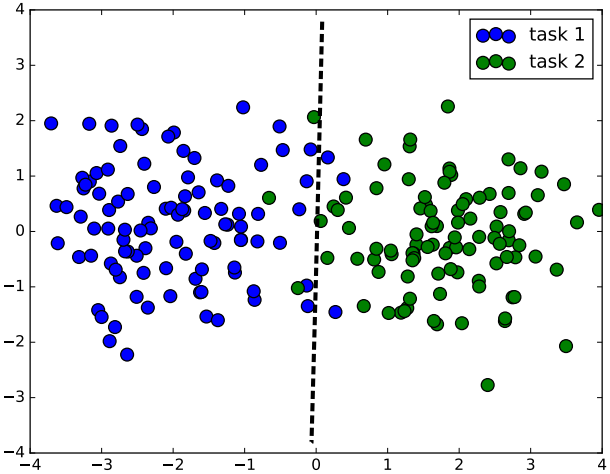
where E is the empirical error of the best classifier in $\mathcal{H}$ on the virtual binary classification problem with

- class 1: all samples of tasks 1
- class 2: all samples of tasks 2

Discrepancy illustration

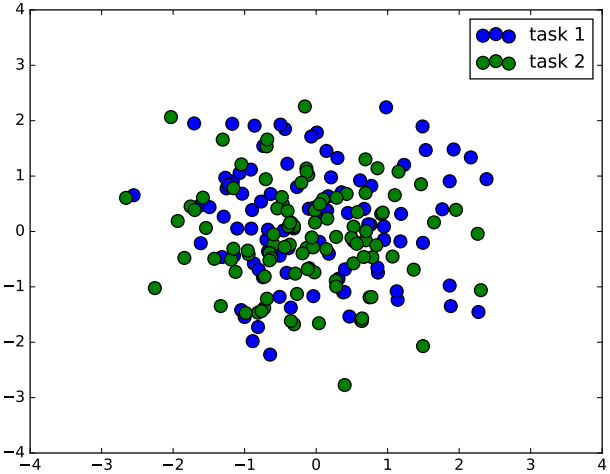


Discrepancy illustration

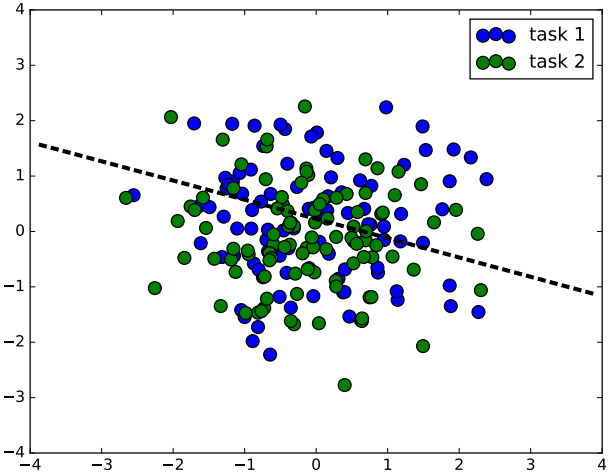


Good separating linear classifier $\rightarrow$ large discrepancy (w.r.t. linear hypotheses class)

Discrepancy illustration

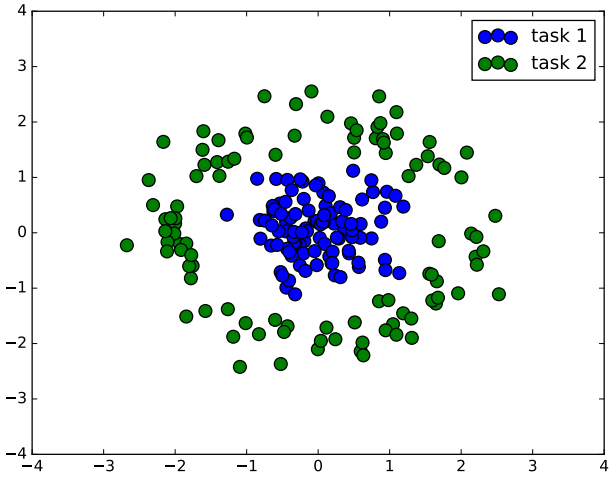


Discrepancy illustration

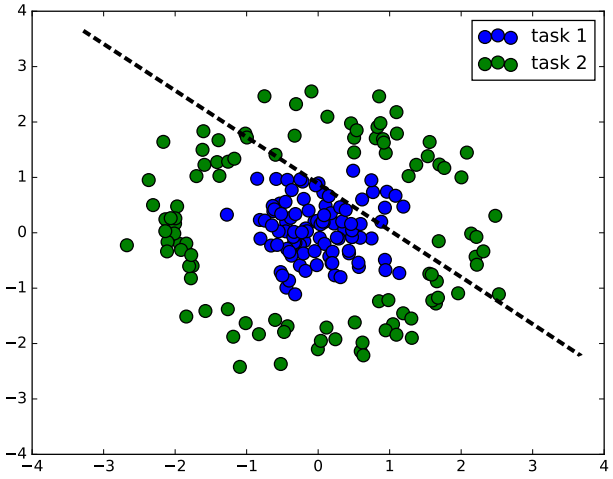


No good separating linear classifier $\rightarrow$ small discrepancy (w.r.t. linear hypotheses class)

Discrepancy illustration

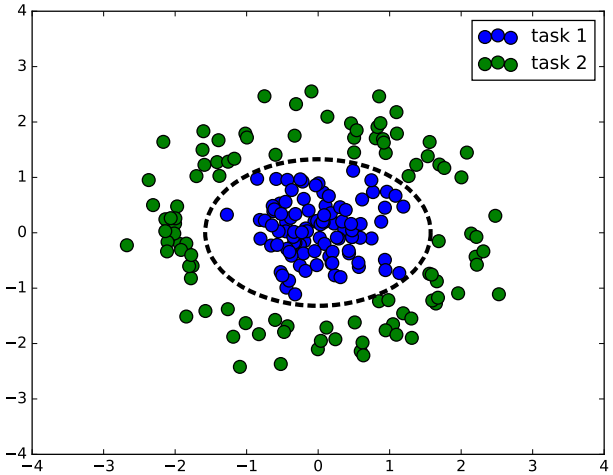


Discrepancy illustration



No good separating linear classifier $\rightarrow$ small discrepancy (w.r.t. linear hypotheses class)

Discrepancy illustration



No good separating linear classifier $\rightarrow$ small discrepancy (w.r.t. linear hypotheses class) but could be large for non-linear hypotheses, e.g., quadratic

Learning classifiers for unlabeled task

Interpret as domain adaptation

- classifier trained from samples of one distribution $\langle D_s, f_s \rangle$
- classifier used for predicting with respect to another distribution $\langle D_t, f_t \rangle$

Discrepancy allows relating the performance of a hypothesis on one task to its performance on a different task:

Domain Adaptation Bound [S. Ben-David, 2010]

For any two tasks $\langle D_s, f_s \rangle$ and $\langle D_t, f_t \rangle$ and any hypothesis $h \in \mathcal{H}$ the following holds:

$$\mathrm{er}_t(h) \leq \mathrm{er}_s(h) + \mathrm{disc}(D_s, D_t) + \lambda_{st},$$

where $\lambda_{st} = \min_{h \in H}(\mathrm{er}_s(h) + \mathrm{er}_t(h))$.

Note: λ_{st} depend on the labeling functions, f_s, f_t, and is not observable.

Active Tasks Selection for Multi-Task Learning with Unlabeled Tasks

Situation:

- $\langle D_1, f_1 \rangle, \ldots, \langle D_T, f_T \rangle$: tasks
- unlabeled $S_1, \ldots, S_T$: sample sets for T tasks, $|S_t| = n$
- for a subset of k tasks we can ask for labeled data, $\overline{S}_t$ with $|\overline{S}_t| = m$
- Goal:
  - identify a subset of k tasks to be labeled, $t_1, \ldots, t_k$,
  - decide between which tasks the classifiers should be shared
  - ask for labeled training sets
  - learn all jointly classifiers

Parameterization:

- assignment vector, $C = (c_1, \ldots, c_T) \in \{1, \ldots, T\}^T$,
  with at most k different values occurring as entries
- $c_s = t$ indicates that we solve tasks s using the classifier from task t

Theorem [Pentina, CHL. 2016]

Let $\delta > 0$. With probability at least $1 - \delta$ the following inequality holds uniformly for all possible assignments $C = (c_1, \ldots, c_T) \in \{1, \ldots, T\}^T$ and all $h \in H$:

$$\frac{1}{T}\sum_{t=1}^{T} \mathrm{er}_t(h_{c_t}) \leq \frac{1}{T}\sum_{t=1}^{T} \widehat{\mathrm{er}}_{c_t}(h_{c_t}) + \frac{1}{T}\sum_{t=1}^{T} \widehat{\mathrm{disc}}(S_t, S_{c_t}) + \frac{1}{T}\sum_{t=1}^{T} \lambda_{tc_t}$$
$$+2\sqrt{\frac{2d\log(2n) + 2\log(T) + \log(4/\delta)}{n}} + \sqrt{\frac{2d\log(em/d)}{m}} + \sqrt{\frac{\log(T) + \log(2/\delta)}{2m}},$$

where

- $\mathrm{er}_t(h) =$ generalization error of h on task t,
- $\widehat{\mathrm{er}}_t(h) =$ training error of h on task t,
- $\lambda_{st} = \mathbf{min}_{h \in \mathcal{H}}(\mathrm{er}_{D_s}(f_s, h) + \mathrm{er}_{D_t}(f_t, h))$

Theorem – Executive Summary

generalization error over all tasks $\leq$

$$\text{training error on labeled tasks} + \sum_{t=1}^{T} \widehat{\mathrm{disc}}(S_t, S_{c_t}) + \frac{1}{T}\sum_{t=1}^{T} \lambda_{tc_t} + \dots$$

Strategy: minimize (computable terms of) the right hand side

Algorithm:

- estimate $\widehat{\mathrm{disc}}(S_i, S_j)$ between all given tasks
- find assignment C that minimizes $\sum_{t=1}^{T} \widehat{\mathrm{disc}}(S_t, S_{c_t})$ $\quad \rightarrow$ k-mediods clustering
- request labels for cluster center tasks and learn classifiers h_{c_t}
- for other tasks, use classifier h_{c_t} of nearest center

Theorem – Executive Summary

generalization error over all tasks $\leq$

$$\text{training error on labeled tasks} + \sum_{t=1}^{T} \widehat{\text{disc}}(S_t, S_{c_t}) + \frac{1}{T}\sum_{t=1}^{T} \lambda_{tc_t} + \dots$$

Strategy: minimize (computable terms of) the right hand side

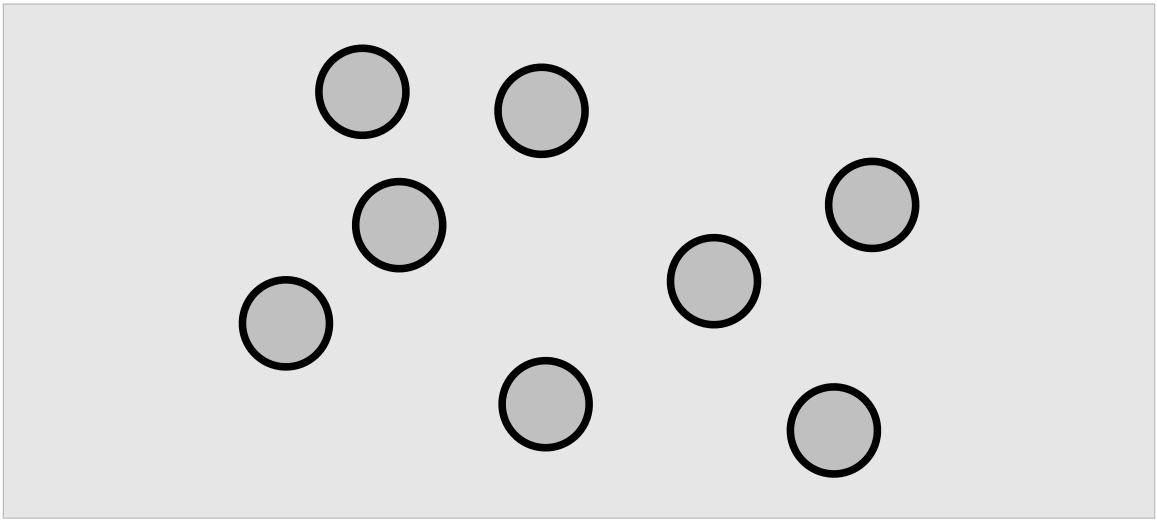
Algorithm:

- estimate $\widehat{\text{disc}}(S_i, S_j)$ between all given tasks
- find assignment C that minimizes $\sum_{t=1}^{T} \widehat{\text{disc}}(S_t, S_{c_t})$ $\quad \rightarrow$ k-mediods clustering
- request labels for cluster center tasks and learn classifiers h_{c_t}
- for other tasks, use classifier h_{c_t} of nearest center

What about λ_{tc_t}?

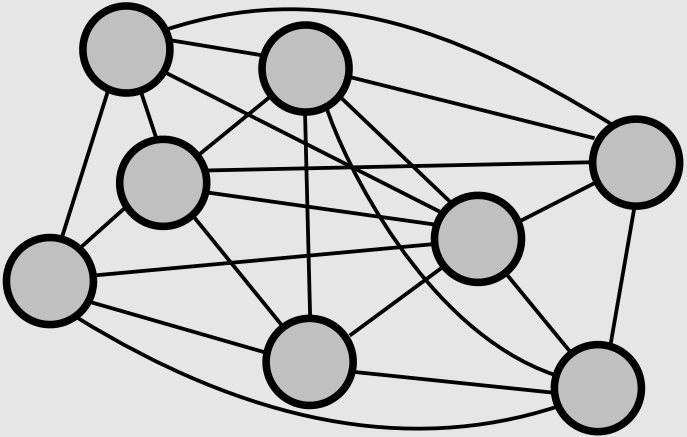
- small under assumption that *"similar tasks have similar labeling functions"*

Active Task Selection for Multi-Task Learning: single-source transfer

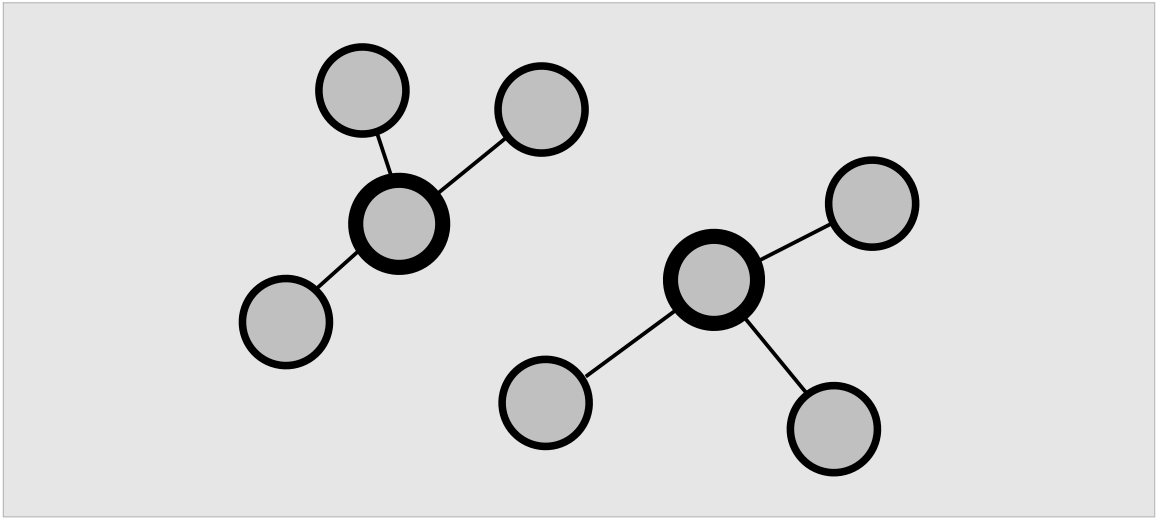


Given: tasks (circles) with only unlabeled data

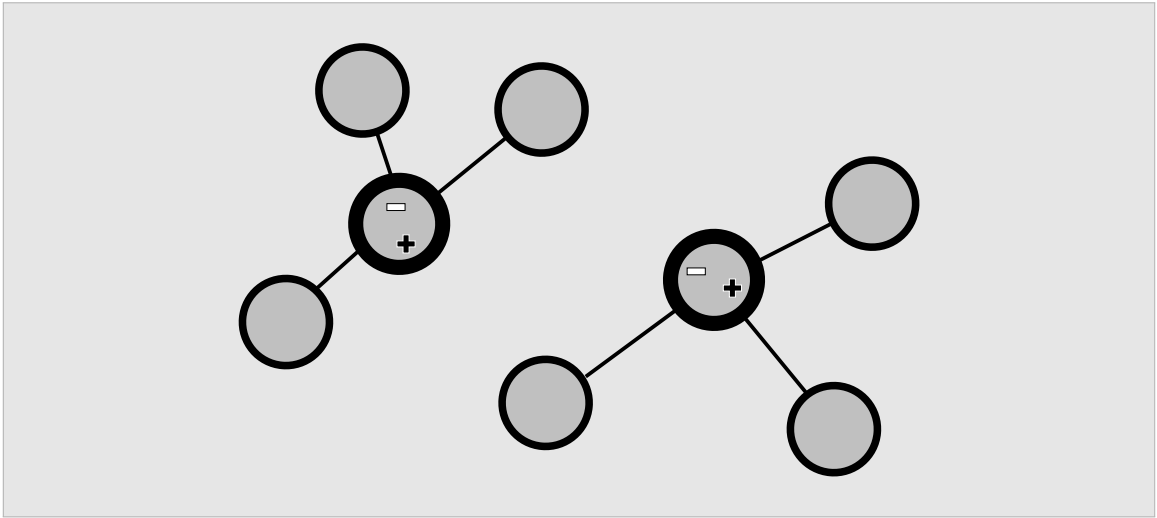
Active Task Selection for Multi-Task Learning: single-source transfer



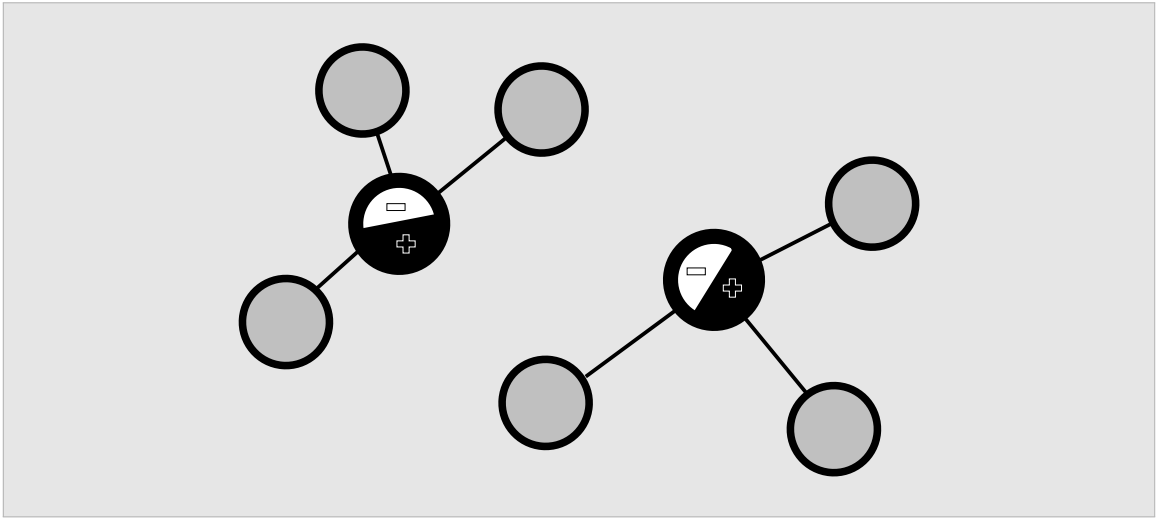
Step 1: compute pairwise discrepancies



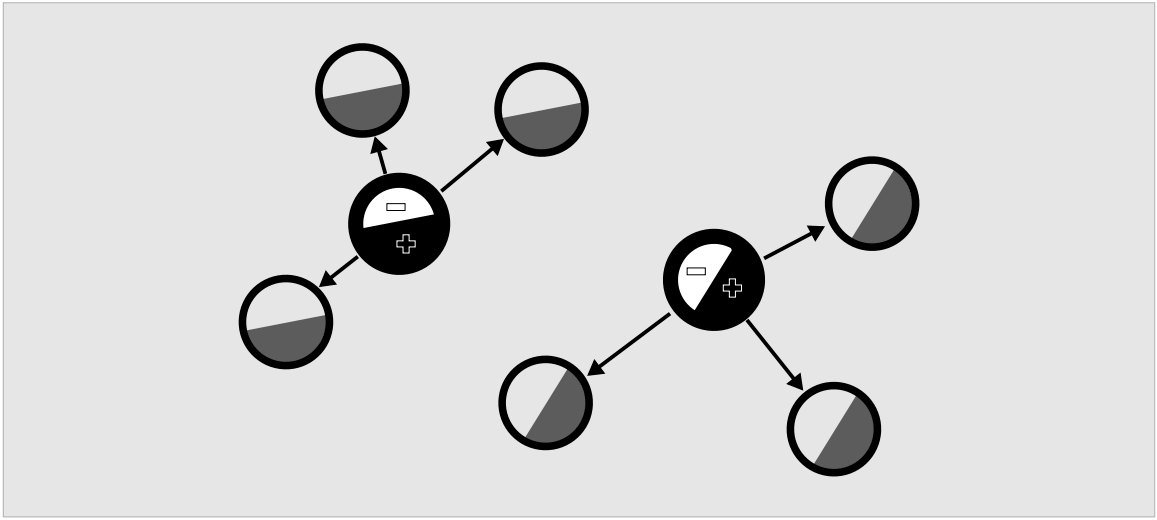
Step 2: cluster tasks by k-mediod



Step 3: request labels for tasks corresponding to cluster centroids



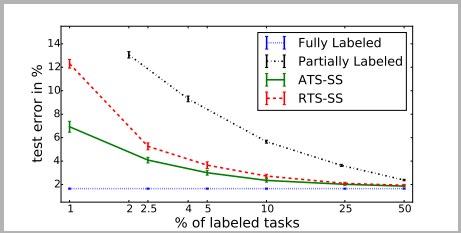
Step 4: learn classifier for centroid tasks



Step 5: transfer classifiers to other tasks in cluster

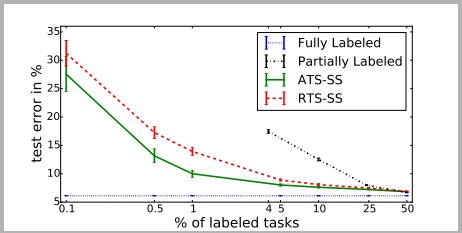
Experiments

Synthetic



1000 tasks (Gaussians in $\mathbb{R}^2$)

ImageNet (ILSVRC2010)



999 tasks ("1 class vs. others")

Methods:

- **ATS-SS**: Active Tasks Selection + discrepancy based transfer
- **RTS-SS**: Random Tasks Selection + discrepancy based transfer

Baselines:

- **Fully Labeled**: all T tasks labeled, 100 labeled examples each
- Partially Labeled: k tasks labeled, $100k/T$ labeled examples each

Extension: can we transfer from more than one task?

New design choice:

- Q: what classifier to use for unlabeled tasks?
- A: train on weighted combination of the training sets of "similar" labeled tasks

Parameterization: weight matrix $\alpha \in \mathbb{R}^{T \times T}$

- α_{st} indicates how much tasks t relies on samples from tasks s
- columns of α sum to 1
- only k non-zero rows of α (indicates which tasks to label)

Note: transfer is now also possible between labeled tasks

Theorem [Pentina, CHL. 2016]

Let $\delta > 0$. Then, provided that the choice of labeled tasks $I = \{i_1, \dots, i_k\}$ and the weights $\alpha^1, \dots, \alpha^T \in \Lambda^I$ are determined by the unlabeled data only, the following inequality holds with probability at least $1-\delta$ for all possible choices of $h_1, \dots, h_T \in H$:

$$\frac{1}{T}\sum_{t=1}^{T}\mathrm{er}_t(h_t) \le \frac{1}{T}\sum_{t=1}^{T}\widehat{\mathrm{er}}_{\alpha^t}(h_t) + \frac{1}{T}\sum_{t=1}^{T}\sum_{i\in I}\alpha_i^t\widehat{\mathrm{disc}}(S_t, S_i) + \frac{A}{T}\|\alpha\|_{2,1} + \frac{B}{T}\|\alpha\|_{1,2} + \frac{1}{T}\sum_{t=1}^{T}\sum_{i\in I}\alpha_i^t\lambda_{ti},$$
$$+ \sqrt{\frac{8(\log T + d\log(enT/d))}{n}} + \sqrt{\frac{2}{n}\log\frac{4}{\delta}} + 2\sqrt{\frac{2d\log(2n) + \log(4T^2/\delta)}{n}}$$

where

$$\|\alpha\|_{2,1} = \sum_{t=1}^{T}\sqrt{\sum_{i\in I}(\alpha_i^t)^2}, \quad \|\alpha\|_{1,2} = \sqrt{\sum_{i\in I}\left(\sum_{t=1}^{T}\alpha_i^t\right)^2}, \quad A = \sqrt{\frac{2d\log(ekm/d)}{m}}, \quad B = \sqrt{\frac{\log(4/\delta)}{2m}}.$$

Theorem – Executive Summary

The generalization error over all T tasks can be bounded uniformly in weights α and $h_1, \ldots, h_T$ by a weighted combination of

- training errors on selected tasks
- discrepancies between tasks
- $\|\alpha\|_{2,1}$ and $\|\alpha\|_{1,2}$

plus λ_{st}-terms (that we assume to be small), plus complexity terms (constants).

Theorem – Executive Summary

The generalization error over all T tasks can be bounded uniformly in weights α and $h_1, \ldots, h_T$ by a weighted combination of

- training errors on selected tasks
- discrepancies between tasks
- $\|\alpha\|_{2,1}$ and $\|\alpha\|_{1,2}$

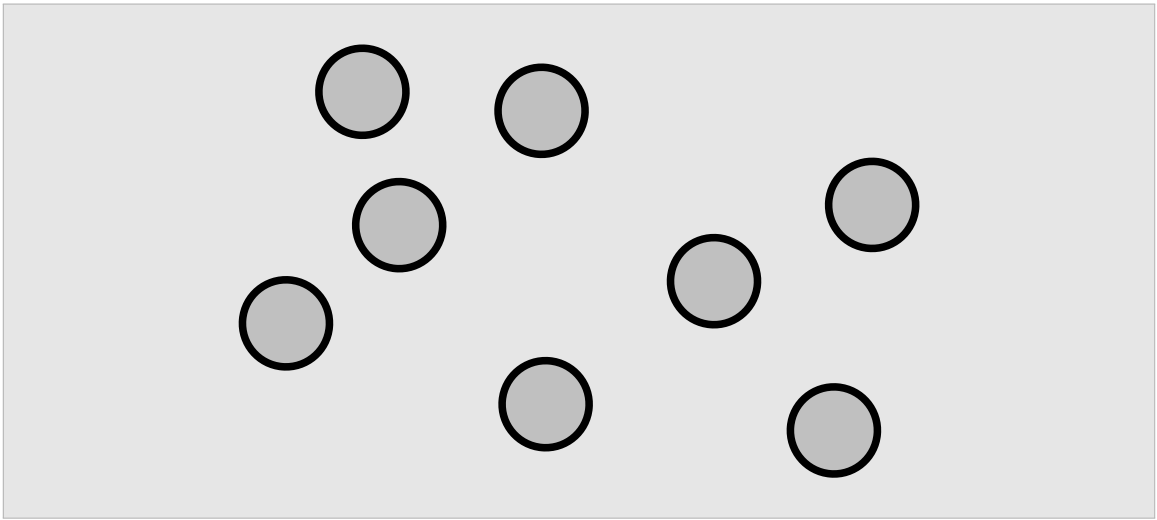
plus λ_{st}-terms (that we assume to be small), plus complexity terms (constants).

Strategy: minimize (computable terms of) right hand side

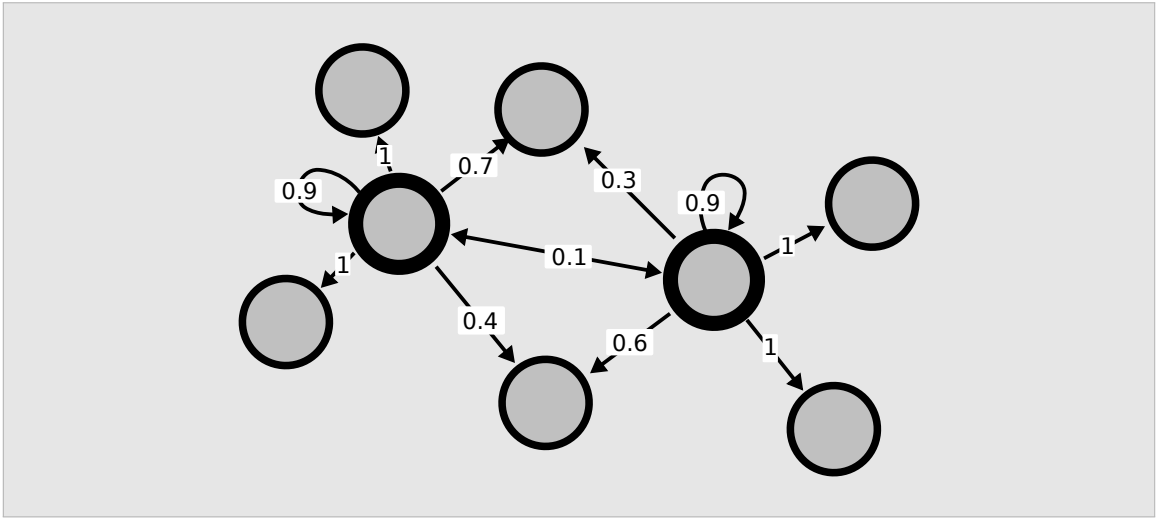
Algorithm:

- estimate $\widehat{\mathrm{disc}}(S_i, S_j)$ between all tasks
- minimize α-weighted discrepancies $+ \|\alpha\|_{2,1} + \|\alpha\|_{1,2}$ w.r.t. α
- request labels for tasks corresponding to non-zero rows of α
- learn classifiers h_t for each task from α-weighted sample sets

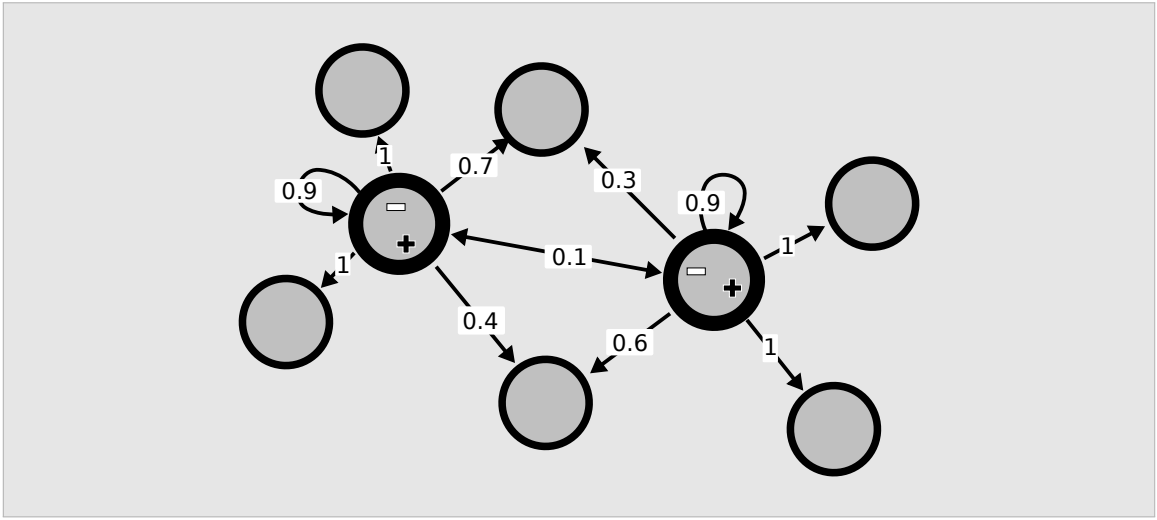
Active Task Selection for Multi-Task Learning: multi-source transfer



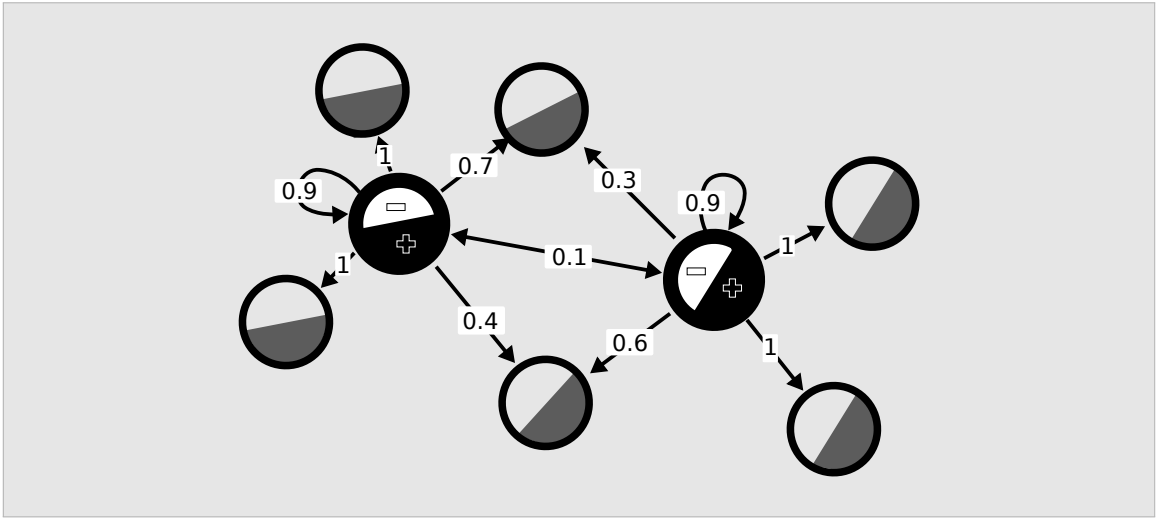
Given: tasks (circles) with only unlabeled data



Step 1: select prototypes and compute amount of sharing



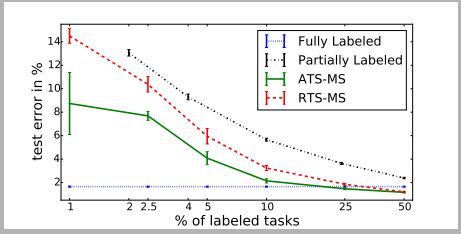
Step 2: request labels for prototypes



Step 3: learn classifier for each task using shared samples

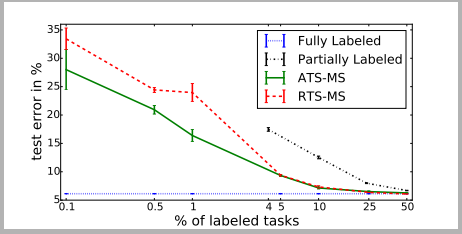
Experiments

Synthetic



1000 tasks (Gaussians in 2D)

ImageNet (ILSVRC2010)



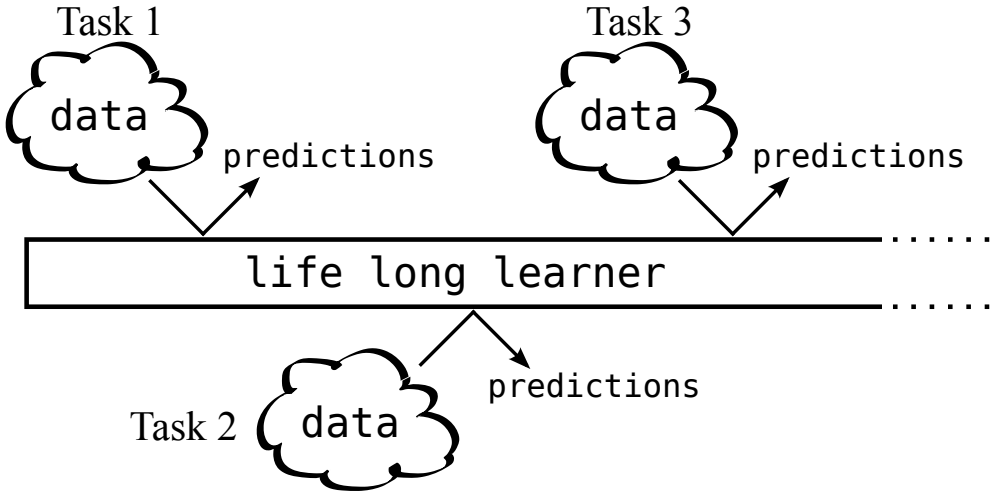
999 tasks ("1 class vs. others")

Methods:

- **ATS-MS**: Active Tasks Selection + transfer
- **RTS-MS**: Random Tasks Selection + transfer

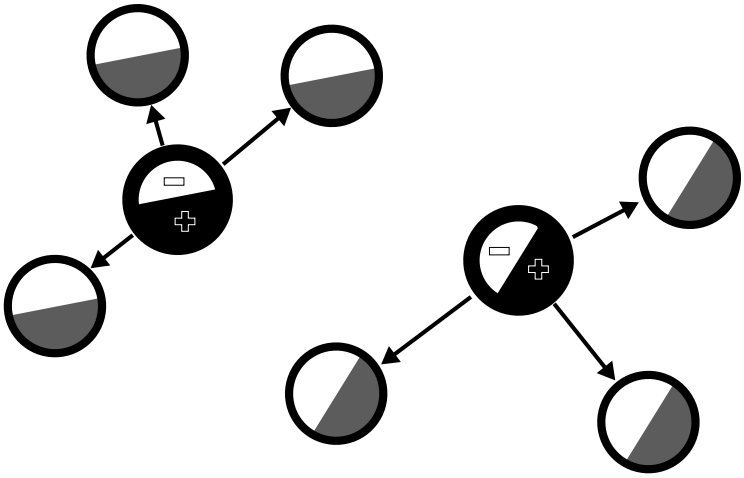
Baselines:

- **Fully Labeled**: all T tasks labeled, 100 labeled examples each
- Partially Labeled: k tasks labeled, $100k/T$ labeled examples each

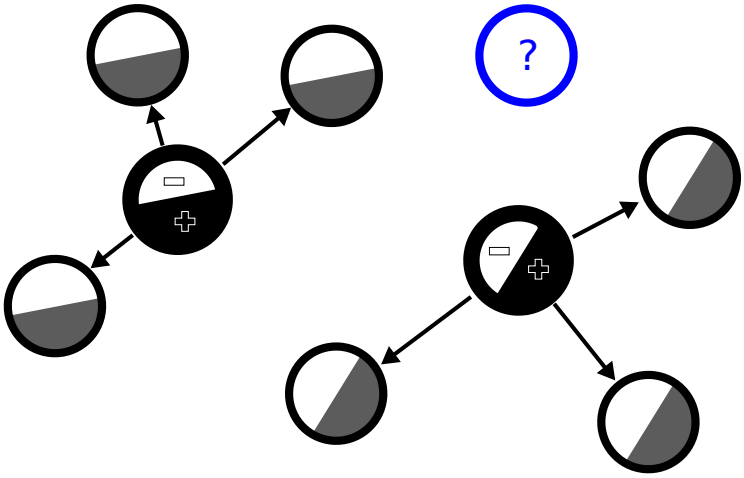


From Multi-Task to Lifelong Learning...

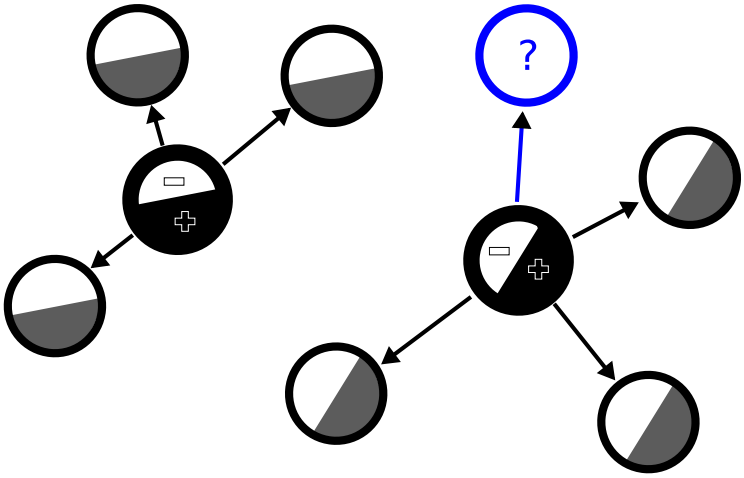
How to handle a new task?



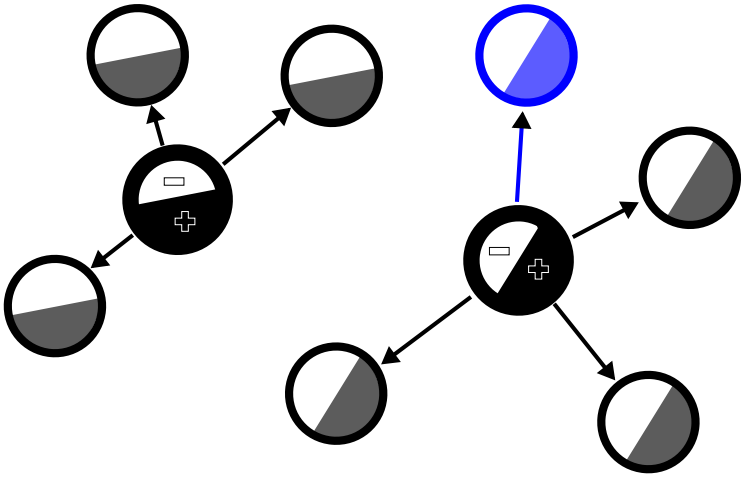
How to handle a new task?



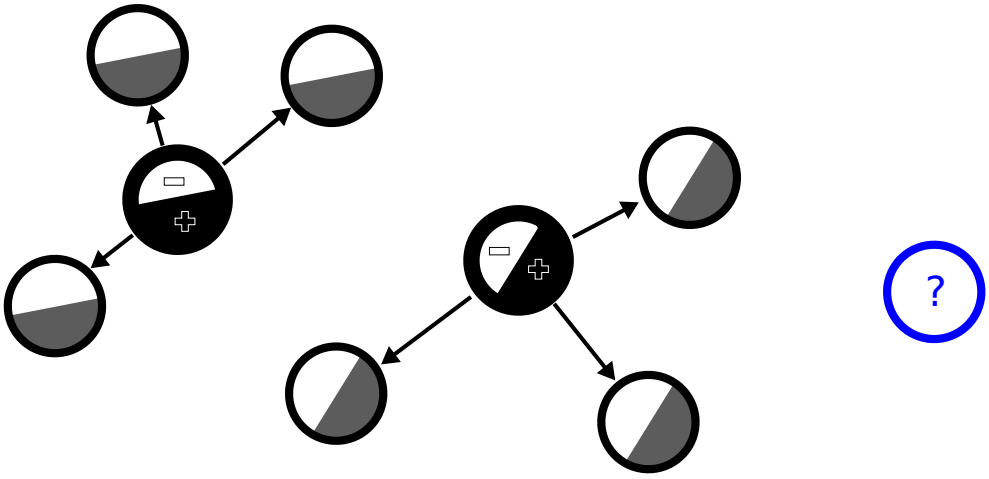
How to handle a new task?



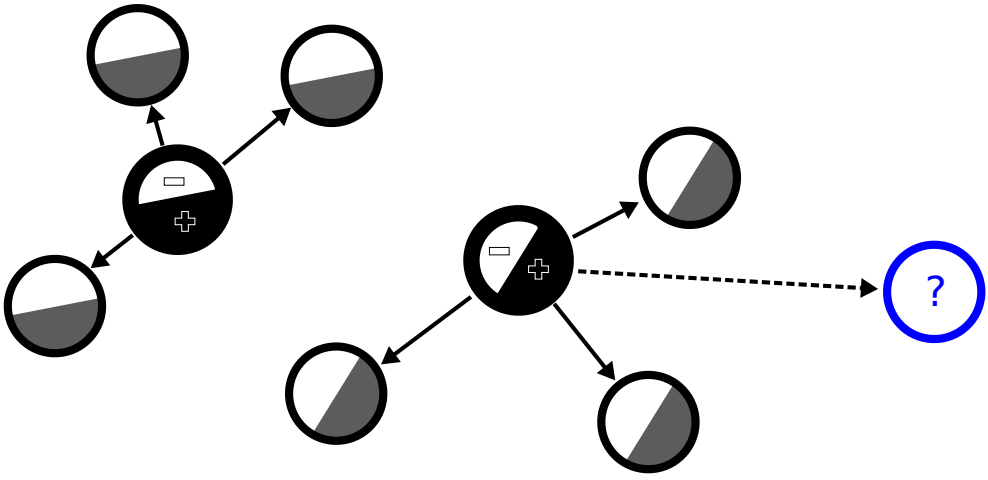
How to handle a new task?



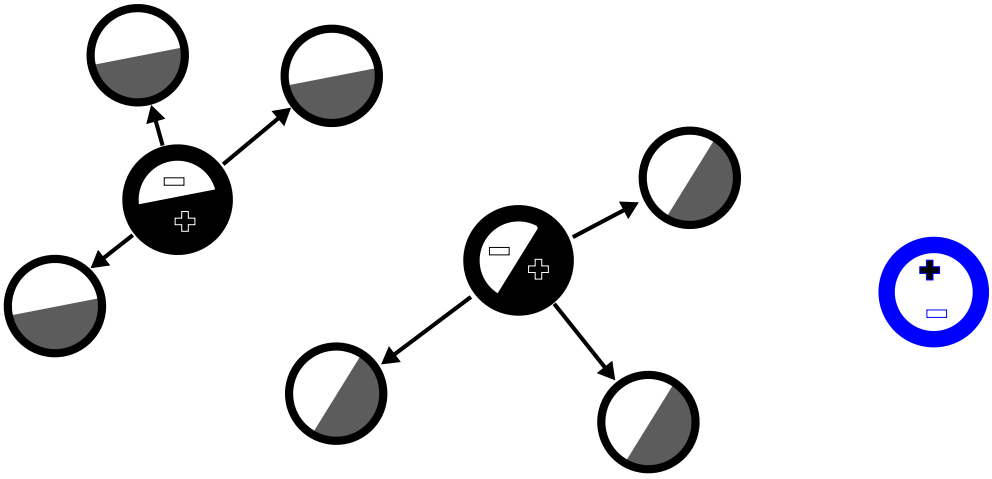
How to handle a new task?



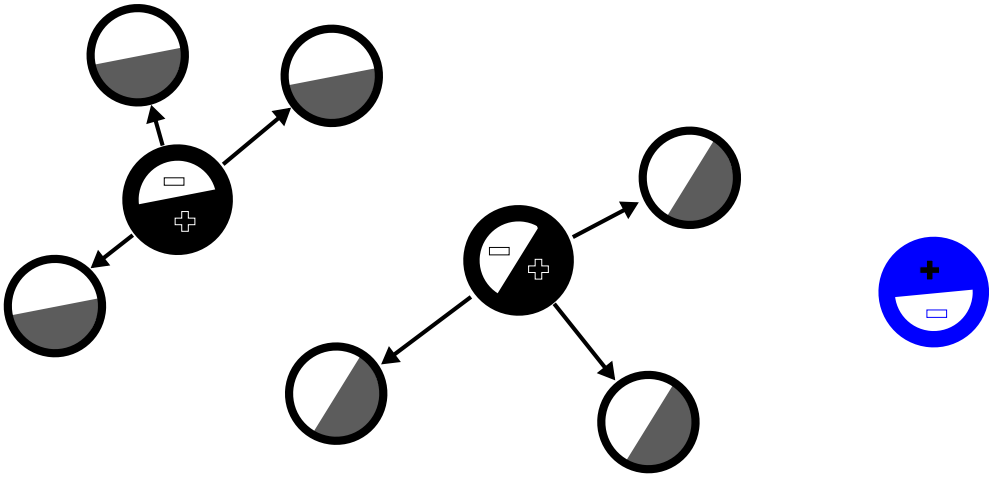
How to handle a new task?



How to handle a new task?



How to handle a new task?



Summary

Multi-Task Learning with Unlabeled Tasks

- many (similar) tasks to solve
- how to avoid having to collect annotation for all of them?

Active Task Selection

- identify most informative tasks and have only those labeled
- transfer information from labeled to unlabeled tasks
- principled algorithms, derived from generalization bounds

Lifelong Learning

- new tasks coming in: use similarity to previous tasks to
  - solve using a previous classifier, or
  - ask for new training annotation

Thank you...