PAC-Bayesian Theory and Domain Adaptation Algorithms

Pascal Germain

November 25, 2015
Plan

1. Basic Definitions

2. PAC-Bayesian Theory
   - Majority Vote Classifiers
   - A Classical PAC-Bayesian Theorem
   - A General PAC-Bayesian Theorem
   - Transductive Learning
   - Rényi-Based Theorem

3. Domain Adaptation Algorithms
   - Ben-David et al.’s Domain Divergence
   - A First PAC-Bayesian Algorithm
   - A Second PAC-Bayesian Algorithm
   - A Neural Network / Representation Learning Algorithm

4. Conclusion and future works
Plan

1. Basic Definitions

2. PAC-Bayesian Theory
   - Majority Vote Classifiers
   - A Classical PAC-Bayesian Theorem
   - A General PAC-Bayesian Theorem
   - Transductive Learning
   - Rényi-Based Theorem

3. Domain Adaptation Algorithms
   - Ben-David et al.’s Domain Divergence
   - A First PAC-Bayesian Algorithm
   - A Second PAC-Bayesian Algorithm
   - A Neural Network / Representation Learning Algorithm

4. Conclusion and future works
Definitions

Learning example

An example \((x, y) \in \mathcal{X} \times \mathcal{Y}\) is a description-label pair.

Data generating distribution

Each example is an observation from distribution \(D\) on \(\mathcal{X} \times \mathcal{Y}\).

Learning sample

\[ S \overset{\text{def}}{=} \{ (x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n) \} \sim D^n \]

Classifier (or hypothesis)

\[ h : \mathcal{X} \to \mathcal{Y} \]

Binary classifier

\[ h : \mathcal{X} \to \{-1, +1\} \]

Learning algorithm

\[ A(S) \rightarrow h \]
I.I.D. Assumption

Examples are generated \textit{i.i.d.} by a distribution $D$ on $\mathcal{X} \times \mathcal{Y}$.

$$S = \{ (x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n) \} \sim D^n$$
Risks

**Risk (or generalization error)**

Probability of misclassifying an example generated by distribution $D$:

$$R_D(h) \overset{\text{def}}{=} \Pr_{(x,y) \sim D} (h(x) \neq y)$$

$$= \mathbb{E}_{(x,y) \sim D} [I[y \cdot h(x) \leq 0]], \quad \langle \text{binary classification} \rangle$$

where $I[a] = 1$ if predicate $a$ is true; $I[a] = 0$ otherwise.

**Empirical risk**

Error rate on the learning sample $S \sim D^n$:

$$\hat{R}_S(h) \overset{\text{def}}{=} \frac{1}{n} \sum_{i=1}^{n} I[y_i \cdot h(x_i) \leq 0].$$
Plan

1 Basic Definitions

2 PAC-Bayesian Theory
   - Majority Vote Classifiers
   - A Classical PAC-Bayesian Theorem
   - A General PAC-Bayesian Theorem
   - Transductive Learning
   - Rényi-Based Theorem

3 Domain Adaptation Algorithms
   - Ben-David et al.’s Domain Divergence
   - A First PAC-Bayesian Algorithm
   - A Second PAC-Bayesian Algorithm
   - A Neural Network / Representation Learning Algorithm

4 Conclusion and future works
PAC-Bayesian Theory

Initiated by David McAllester (1999), the PAC-Bayesian theory gives generalization guarantees on majority votes of classifiers.

**PAC guarantees (Probably Approximately Correct)**

With probability at least $\approx 1 - \delta$, the risk of classifier $h$ is less than $\approx \epsilon$

$$\Pr_{S \sim D^n} \left( R_D(h) \leq \epsilon(R_S(h), n, \ldots) \right) \geq 1 - \delta$$

**Bayesian flavor**

Incorporates a priori knowledge about the learning problem as a probability distribution over a family of classifiers.

**Training bounds**

- Gives generalization guarantees not based on testing sample;
- Inspiration for conceiving new learning algorithms.
Plan

1. Basic Definitions

2. PAC-Bayesian Theory
   - Majority Vote Classifiers
   - A Classical PAC-Bayesian Theorem
   - A General PAC-Bayesian Theorem
   - Transductive Learning
   - Rényi-Based Theorem

3. Domain Adaptation Algorithms
   - Ben-David et al.’s Domain Divergence
   - A First PAC-Bayesian Algorithm
   - A Second PAC-Bayesian Algorithm
   - A Neural Network / Representation Learning Algorithm

4. Conclusion and future works
Majority Vote Classifiers

Given:
- A set of voters $\mathcal{H} = \{ h_1, h_2, h_3, \ldots \}$ (discrete or continuous);
- A weight distribution $Q$ on $\mathcal{H}$.

Weighted majority vote

To predict the label of $x \in \mathcal{X}$, the classifier asks for the prevailing opinion

$$B_Q(x) \overset{\text{def}}{=} \text{sgn} \left( \mathbb{E}_{h \sim Q} h(x) \right)$$

Many learning algorithms output majority vote classifiers

AdaBoost, Random Forests, Bagging, ...
A Surrogate Loss

Given

- A data distribution \( D \) on \( \mathcal{X} \times \{-1, +1\} \);
- A weight distribution \( Q \) on the set of voters \( \mathcal{H} \).

Majority vote risk (or Bayes Risk)

\[
R_D(B_Q) \overset{\text{def}}{=} \mathbb{E}_{(x,y) \sim D} \left[ \mathbb{E}_{h \sim Q} y \cdot h(x) \leq 0 \right]
\]

Gibbs Risk

The stochastic Gibbs classifier \( G_Q(x) \) draws \( h' \in \mathcal{H} \) according to \( Q \) and output \( h'(x) \).

\[
R_D(G_Q) \overset{\text{def}}{=} \mathbb{E}_{(x,y) \sim D} \mathbb{E}_{h \sim Q} \left[ y \cdot h(x) \leq 0 \right] = \mathbb{E}_{(x,y) \sim D} \left( \frac{1}{2} - \frac{1}{2} \mathbb{E}_{h \sim Q} y \cdot h(x) \right)
\]

Factor two

It is well-known that

\[
R_D(B_Q) \leq 2 \times R_D(G_Q)
\]
From the **Factor 2** to the $C$-bound

From Markov’s inequality ($Pr(X \geq a) \leq \frac{EX}{a}$), we obtain:

**Factor 2 bound**

$$R_D(B_Q) = \Pr_{(x,y)\sim D} \left( 1 - y \cdot h(x) \geq 1 \right) \leq \mathbb{E}_{(x,y)\sim D} \left( 1 - y \cdot h(x) \right) = 2R_D(G_Q).$$

From Chebyshev’s inequality ($Pr(X - \mathbb{E}X \geq a) \leq \frac{\text{Var} X}{a^2 + \text{Var} X}$), we obtain:

**The $C$-bound (Lacasse et al., 2006)**

$$R_D(B_Q) \leq C^D_Q \overset{\text{def}}{=} 1 - \left( 1 - 2 \cdot R_D(G_Q) \right)^2$$

where $d^D_Q$ is the **expected disagreement**:

$$d^D_Q \overset{\text{def}}{=} \mathbb{E}_{(x,\cdot)\sim D} \mathbb{E}_{h_1\sim Q} \mathbb{E}_{h_2\sim Q} \mathbb{I} \left[ h_1(x) \neq h_2(x) \right] = \frac{1}{2} \left( 1 - \mathbb{E}_{(x,\cdot)\sim D'} \left[ \mathbb{E}_{h\sim Q} h(x)^2 \right] \right)$$
Plan

1. Basic Definitions

2. PAC-Bayesian Theory
   - Majority Vote Classifiers
   - A Classical PAC-Bayesian Theorem
   - A General PAC-Bayesian Theorem
   - Transductive Learning
   - Rényi-Based Theorem

3. Domain Adaptation Algorithms
   - Ben-David et al.’s Domain Divergence
   - A First PAC-Bayesian Algorithm
   - A Second PAC-Bayesian Algorithm
   - A Neural Network / Representation Learning Algorithm

4. Conclusion and future works
A Classical PAC-Bayesian Theorem

Two principal components

- The **Gibbs empirical risk**:
  \[
  \hat{R}_S(G_Q) \overset{\text{def}}{=} \frac{1}{n} \sum_{i=1}^{n} \left( \frac{1}{2} - \frac{1}{2} \mathbb{E}_{h \sim Q} y_i \cdot h(x_i) \right)
  \]

- The **Kullback-Leibler divergence** between the prior \(P\) and the posterior \(Q\):
  \[
  \text{KL}(Q\|P) \overset{\text{def}}{=} \mathbb{E}_{h \sim Q} \ln \frac{Q(h)}{P(h)}
  \]

PAC-Bayesian theorem (McAllester, 2003)

For any distribution \(D\) on \(\mathcal{X} \times \mathcal{Y}\), for any set of voters \(\mathcal{H}\), for any distribution \(P\) on \(\mathcal{H}\), for any \(\delta \in (0,1]\), we have, with probability at least \(1-\delta\) over the choice of \(S \sim D^n\),

\[
\forall Q \text{ on } \mathcal{H} : \quad R_D(G_Q) \leq \hat{R}_S(G_Q) + \sqrt{\frac{1}{2n} \left[ \text{KL}(Q\|P) + \ln \frac{2\sqrt{n}}{\delta} \right]}
\]
Plan

1. Basic Definitions

2. PAC-Bayesian Theory
   - Majority Vote Classifiers
   - A Classical PAC-Bayesian Theorem
   - A General PAC-Bayesian Theorem
   - Transductive Learning
   - Rényi-Based Theorem

3. Domain Adaptation Algorithms
   - Ben-David et al.’s Domain Divergence
   - A First PAC-Bayesian Algorithm
   - A Second PAC-Bayesian Algorithm
   - A Neural Network / Representation Learning Algorithm

4. Conclusion and future works
A General PAC-Bayesian Theorem

\[ \Delta \text{-function : «distance» between } \hat{R}_S(G_Q) \text{ et } R_D(G_Q) \]

Convex function \( \Delta : [0, 1] \times [0, 1] \rightarrow \mathbb{R} \).

General theorem

For any distribution \( D \) on \( \mathcal{X} \times \mathcal{Y} \), for any set \( \mathcal{H} \) of voters, for any distribution \( P \) on \( \mathcal{H} \), for any \( \delta \in (0, 1] \), and for any \( \Delta \)-function, we have, with probability at least \( 1 - \delta \) over the choice of \( S \sim D^n \),

\[ \forall Q \text{ on } \mathcal{H} : \quad \Delta \left( \hat{R}_S(G_Q), R_D(G_Q) \right) \leq \frac{1}{n} \left[ \text{KL}(Q\|P) + \ln \frac{I_\Delta(n)}{\delta} \right], \]

où

\[ I_\Delta(n) \overset{\text{def}}{=} \sup_{r \in [0, 1]} \left[ \sum_{k=0}^{n} \text{Bin}(k; n, r) e^{n\Delta(k/n, r)} \right] \]
**General theorem**

\[
\Pr_{S \sim D^n} \left( \forall Q \text{ on } \mathcal{H} : \Delta \left( \hat{R}_S(G_Q), R_D(G_Q) \right) \leq \frac{1}{n} \left[ \text{KL}(Q\|P) + \ln \frac{I_{\Delta}(n)}{\delta} \right] \right) \geq 1 - \delta.
\]

**Interpretation.**
General theorem

\[
\Pr_{S \sim D^n} \left( \forall Q \text{ on } \mathcal{H} : \Delta \left( \hat{R}_S(G_Q), R_D(G_Q) \right) \leq \frac{1}{n} \left[ \text{KL}(Q\|P) + \ln \frac{\mathcal{I}_\Delta(n)}{\delta} \right] \right) \geq 1-\delta.
\]

Proof ideas.

Change of Measure Inequality

For any \(P\) and \(Q\) on \(\mathcal{H}\), and for any measurable function \(\phi : \mathcal{H} \to \mathbb{R}\), we have

\[
\mathbb{E}_{h \sim Q} \phi(h) \leq \text{KL}(Q\|P) + \ln \left( \mathbb{E}_{h \sim P} e^{\phi(h)} \right).
\]

Markov’s inequality

\[
\Pr(X \geq a) \leq \frac{\mathbb{E}X}{a} \iff \Pr(X \leq \frac{\mathbb{E}X}{\delta}) \geq 1-\delta.
\]

Probability of observing \(k\) misclassifications among \(n\) examples

Given a voter \(h\), consider a \textit{binomial variable} of \(n\) trials with \textit{success} \(R_D(h)\):

\[
\text{Bin}(k; n, R_D(h)) \overset{\text{def}}{=} \Pr_{S \sim D^n} \left( \hat{R}_S(h) = \frac{k}{n} \right) = \binom{n}{k} \left( R_D(h) \right)^k \left( 1 - R_D(h) \right)^{n-k}.
\]
General theorem

$$\Pr_{S \sim D^n} \left( \forall Q \text{ on } \mathcal{H} : \Delta \left( \hat{R}_S(G_Q), R_D(G_Q) \right) \leq \frac{1}{n} \left[ \text{KL}(Q\|P) + \ln \frac{\mathcal{I}_\Delta(n)}{\delta} \right] \right) \geq 1 - \delta.$$  

Proof.

$$n \cdot \Delta \left( \mathbb{E}_{h \sim Q} \hat{R}_S(h), \mathbb{E}_{h \sim Q} R_D(h) \right) \leq \mathbb{E}_{h \sim Q} n \cdot \Delta \left( \hat{R}_S(h), R_D(h) \right) \leq \text{KL}(Q\|P) + \ln \mathbb{E}_{h \sim P} e^{n \Delta \left( \hat{R}_S(h), R_D(h) \right)}$$

Jensen’s Inequality

$$\leq 1 - \delta \quad \text{KL}(Q\|P) + \ln \frac{1}{\delta} \mathbb{E}_{S' \sim D^n} \mathbb{E}_{h \sim P} e^{n \cdot \Delta \left( R_{S'}(h), R_D(h) \right)}$$

Change of measure

$$\leq 1 - \delta \quad \text{KL}(Q\|P) + \ln \frac{1}{\delta} \mathbb{E}_{h \sim P} \sum_{k=0}^{n} \text{Bin}(k; n, R_D(h)) e^{n \cdot \Delta \left( \frac{k}{n}, R_D(h) \right)}$$

Markov’s Inequality

$$= \text{KL}(Q\|P) + \ln \frac{1}{\delta} \mathbb{E}_{h \sim P} \sup_{r \in [0,1]} \left[ \sum_{k=0}^{n} \text{Bin}(k; n, r) e^{n \Delta \left( \frac{k}{n}, r \right)} \right]$$

Expectation swap

$$\leq \text{KL}(Q\|P) + \ln \frac{1}{\delta} \mathcal{I}_\Delta(n).$$

Binomial law

$$\sup_{r \in [0,1]} \left[ \sum_{k=0}^{n} \text{Bin}(k; n, r) e^{n \Delta \left( \frac{k}{n}, r \right)} \right]$$

Supremum over risk

$$\leq \mathcal{I}_\Delta(n).$$
General theorem

\[
\Pr_{S \sim D^n} \left( \forall Q \text{ on } \mathcal{H} : \Delta \left( \hat{R}_S(G_Q), R_D(G_Q) \right) \leq \frac{1}{n} \left[ \text{KL}(Q \| P) + \ln \frac{I_\Delta(n)}{\delta} \right] \right) \geq 1 - \delta.
\]

Corollary

[...] with probability at least 1−δ over the choice of S ∼ D^n,

\forall Q \text{ on } \mathcal{H} :

(a) \( \text{kl} \left( \hat{R}_S(G_Q), R_D(G_Q) \right) \leq \frac{1}{n} \left[ \text{KL}(Q \| P) + \ln \frac{2\sqrt{n}}{\delta} \right] \), (Langford and Seeger, 2001)

(b) \( R_D(G_Q) \leq \hat{R}_S(G_Q) + \sqrt{\frac{1}{2n} \left[ \text{KL}(Q \| P) + \ln \frac{2\sqrt{n}}{\delta} \right]} \), (McAllester, 1999)

(c) \( R_D(G_Q) \leq \frac{1}{1 - e^{-c}} \left( c \cdot \hat{R}_S(G_Q) + \frac{1}{n} \left[ \text{KL}(Q \| P) + \ln \frac{1}{\delta} \right] \right) \). (Catoni, 2007)

\[ \text{kl}(q, p) \overset{\text{def}}{=} q \ln \frac{q}{p} + (1 - q) \ln \frac{1 - q}{1 - p} \geq 2(q - p)^2, \]

\[ \Delta_c(q, p) \overset{\text{def}}{=} - \ln \left[ 1 - (1 - e^{-c}) \cdot p \right] - c \cdot q. \]
Bounding the Expected Disagreement

**Expected disagreement**

\[
    d^D_Q \overset{\text{def}}{=} \mathbb{E}_{(x,\cdot) \sim D} \mathbb{E}_{h_1 \sim Q} \mathbb{E}_{h_2 \sim Q} \mathbb{I} \left[ h_1(x) \neq h_2(x) \right] = \frac{1}{2} \left( 1 - \mathbb{E}_{(x,\cdot) \sim D'} \left[ \mathbb{E}_{h \sim Q} h(x) \right]^2 \right)
\]

**General theorem**

[...] with probability at least \(1 - \delta\) over the choice of \(S \sim D^n\),

\[
    \forall Q \text{ on } \mathcal{H} : \quad \Delta \left( d^S_Q, d^D_Q \right) \leq \frac{1}{n} \left[ 2 \text{KL}(Q\|P) + \ln \frac{\mathcal{I}_\Delta(n)}{\delta} \right],
\]

**Corollary**

(a) \( \text{kl} \left( d^S_Q, d^D_Q \right) \leq \frac{1}{n} \left[ 2 \text{KL}(Q\|P) + \ln \frac{2\sqrt{n}}{\delta} \right], \)

(b) \( d^D_Q \leq d^S_Q + \sqrt{\frac{1}{2n} \left[ 2 \text{KL}(Q\|P) + \ln \frac{2\sqrt{n}}{\delta} \right]}, \)

(c) \( d^D_Q \leq \frac{1}{1-e^{-c}} \left( c \cdot d^S_Q + \frac{1}{n} \left[ 2 \text{KL}(Q\|P) + \ln \frac{1}{\delta} \right] \right) \).
Plan

1. Basic Definitions

2. PAC-Bayesian Theory
   - Majority Vote Classifiers
   - A Classical PAC-Bayesian Theorem
   - A General PAC-Bayesian Theorem
   - Transductive Learning
     - Rényi-Based Theorem

3. Domain Adaptation Algorithms
   - Ben-David et al.'s Domain Divergence
   - A First PAC-Bayesian Algorithm
   - A Second PAC-Bayesian Algorithm
   - A Neural Network / Representation Learning Algorithm

4. Conclusion and future works
Transductive Learning

Assumption

Examples are drawn *without replacement* from a finite set $Z$ of size $N$.

\[
S = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\} \subset Z
\]
\[
U = \{(x_{n+1}, \cdot), (x_{n+2}, \cdot), \ldots, (x_N, \cdot)\} = Z \setminus S
\]
General Theorem for Transductive Learning

**Observation**

Inductive learning: \( n \) draws with replacement according to \( D \Rightarrow \) Binomial law.

Transductive learning: \( n \) draws without replacement in \( Z \Rightarrow \) Hypergeometric law.

**Theorem**

For any set \( Z \) of \( N \) examples, for any set \( \mathcal{H} \) of voters, for any distribution \( P \) on \( \mathcal{H} \), for any \( \delta \in (0, 1] \), and for any \( \Delta \)-function, we have, with probability at least \( 1 - \delta \) over the choice of \( n \) examples among \( Z \),

\[
\forall Q \text{ on } \mathcal{H} : \quad \Delta(\hat{R}_S(G_Q), \hat{R}_Z(G_Q)) \leq \frac{1}{n} \left[ \text{KL}(Q \| P) + \ln \frac{T_\Delta(n, N)}{\delta} \right],
\]

where

\[
T_\Delta(n, N) \overset{\text{def}}{=} \max_{K=0 \ldots N} \left[ \sum_{k \in \mathcal{K}_{n, N, K}} \frac{\binom{K}{k} \binom{N-K}{n-k} }{\binom{N}{n}} e^{n \Delta(\frac{k}{n}, \frac{K}{N})} \right],
\]

and \( \mathcal{K}_{n, N, K} \overset{\text{def}}{=} \{ \max[0, K+n-N], \ldots, \min[n, K] \} \).
Theorem

\[
\Pr_{S \sim [Z]^n} \left( \forall Q \text{ on } \mathcal{H} : \Delta \left( \hat{R}_S(G_Q), \hat{R}_Z(G_Q) \right) \leq \frac{1}{n} \left[ \text{KL}(Q \| P) + \ln \frac{T_{\Delta}(n, N)}{\delta} \right] \right) \geq 1 - \delta.
\]

Proof.

\[
n \cdot \Delta \left( E_{h \sim Q} \hat{R}_S(h), E_{h \sim Q} \hat{R}_Z(h) \right)
\leq E_{h \sim Q} n \cdot \Delta \left( \hat{R}_S(h), \hat{R}_Z(h) \right)
\leq \text{KL}(Q \| P) + \ln \frac{1}{\delta} E_{h \sim P} E_{S' \sim [Z]^n} e^{n \cdot \Delta(R_{S'}(h), \hat{R}_Z(h))}
\leq 1 - \delta \text{KL}(Q \| P) + \ln \frac{1}{\delta} E_{h \sim P} \sum_{S' \sim [Z]^n} e^{n \cdot \Delta(R_{S'}(h), \hat{R}_Z(h))}
= \text{KL}(Q \| P) + \ln \frac{1}{\delta} E_{h \sim P} \sum_{k \in K, n, N, R_Z(h)} \frac{\binom{N}{k} \binom{N - N \cdot \hat{R}_Z(h)}{n - k}}{\binom{N}{n}} e^{n \cdot \Delta(k, \hat{R}_Z(h))}
\leq \text{KL}(Q \| P) + \ln \frac{1}{\delta} \max_{K = 0 \ldots N} \left[ \sum_{k \in K, n, N, K} \binom{K}{k} \binom{N - K}{n - k} \frac{1}{\binom{N}{n}} e^{n \Delta(k, K)} \right]
= \text{KL}(Q \| P) + \ln \frac{1}{\delta} T_{\Delta}(n, N).
\]
A New Transductive Bound for the Gibbs Risk

**Corollary**

... with probability at least \(1 - \delta\) over the choice of \(n\) examples among \(Z\),

\[ \forall Q \text{ on } H : \hat{R}_Z(G_Q) \leq \hat{R}_S(G_Q) + \sqrt{\frac{1 - \frac{n}{N}}{2n} \left[ \text{KL}(Q||P) + \ln \frac{3\ln(n)\sqrt{n(1 - \frac{n}{N})}}{\delta} \right]} \]

**Theorem (Derbeko et al., 2004)**

\[ \hat{R}_Z(G_Q) \leq \hat{R}_S(G_Q) + \sqrt{\frac{1 - \frac{n}{N}}{2(n-1)} \left[ \text{KL}(Q||P) + \ln \frac{n(N+1)^7}{\delta} \right]} \]
A New Transductive Bound for the Bayes Risk

Majority Vote Bound

[...] with probability at least $1 - \delta$ over the choice of $n$ examples among $Z$,

\[ \forall Q \text{ on } \mathcal{H} : \]

(a) \[ \hat{R}_Z(B_Q) \leq 2 \times \bar{r} \quad \text{(Factor 2)} \]

(b) \[ \hat{R}_Z(B_Q) \leq 1 - \left( \frac{1 - 2 \times \bar{r}}{1 - 2 \times d_Q^Z} \right)^2 \quad \text{(C-bound)} \]

where

\[ \bar{r} := \hat{R}_S(G_Q) + \sqrt{\frac{1 - \frac{n}{N}}{2n} \left[ \text{KL}(Q\|P) + \ln \frac{3 \ln(n) \sqrt{n(1 - \frac{n}{N})}}{\delta} \right]} , \]

\[ d_Q^Z = \frac{1}{2} \left( 1 - \sum_{i=1}^{N} \mathbb{E}_{h \sim Q} \left[ h(x_i) \right] \right)^2 . \]
Empirical Comparison
Majority votes of *decision stumps* obtained with *AdaBoost*.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>N</th>
<th>n/N</th>
<th>(R_s(B_Q))</th>
<th>Factor 2</th>
<th>C-bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>car</td>
<td>1728</td>
<td>0.1</td>
<td>0.105</td>
<td>1.092</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>1728</td>
<td>0.5</td>
<td>0.115</td>
<td>0.830</td>
<td>0.819</td>
</tr>
<tr>
<td>letter_AB</td>
<td>1555</td>
<td>0.1</td>
<td>0.000</td>
<td><strong>0.914</strong></td>
<td>0.961</td>
</tr>
<tr>
<td></td>
<td>1555</td>
<td>0.5</td>
<td>0.000</td>
<td>0.797</td>
<td>0.626</td>
</tr>
<tr>
<td>mushroom</td>
<td>8124</td>
<td>0.1</td>
<td>0.000</td>
<td><strong>0.964</strong></td>
<td>0.966</td>
</tr>
<tr>
<td></td>
<td>8124</td>
<td>0.5</td>
<td>0.000</td>
<td>0.875</td>
<td>0.546</td>
</tr>
<tr>
<td>nursery</td>
<td>12959</td>
<td>0.1</td>
<td>0.009</td>
<td>0.798</td>
<td>0.692</td>
</tr>
<tr>
<td></td>
<td>12959</td>
<td>0.5</td>
<td>0.010</td>
<td>0.711</td>
<td>0.379</td>
</tr>
<tr>
<td>optdigits</td>
<td>3823</td>
<td>0.1</td>
<td>0.000</td>
<td>1.055</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>3823</td>
<td>0.5</td>
<td>0.026</td>
<td>0.917</td>
<td>0.793</td>
</tr>
<tr>
<td>pageblock</td>
<td>5473</td>
<td>0.1</td>
<td>0.048</td>
<td><strong>0.979</strong></td>
<td>0.992</td>
</tr>
<tr>
<td></td>
<td>5473</td>
<td>0.5</td>
<td>0.057</td>
<td>0.894</td>
<td>0.697</td>
</tr>
<tr>
<td>pendigits</td>
<td>7494</td>
<td>0.1</td>
<td>0.023</td>
<td><strong>0.989</strong></td>
<td>0.997</td>
</tr>
<tr>
<td></td>
<td>7494</td>
<td>0.5</td>
<td>0.041</td>
<td>0.912</td>
<td>0.706</td>
</tr>
<tr>
<td>segment</td>
<td>2310</td>
<td>0.1</td>
<td>0.000</td>
<td>1.101</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>2310</td>
<td>0.5</td>
<td>0.014</td>
<td>0.920</td>
<td>0.834</td>
</tr>
<tr>
<td>spambase</td>
<td>4601</td>
<td>0.1</td>
<td>0.115</td>
<td>1.096</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>4601</td>
<td>0.5</td>
<td>0.137</td>
<td>0.973</td>
<td>0.961</td>
</tr>
</tbody>
</table>
Plan

1. Basic Definitions

2. PAC-Bayesian Theory
   - Majority Vote Classifiers
   - A Classical PAC-Bayesian Theorem
   - A General PAC-Bayesian Theorem
   - Transductive Learning
   - Rényi-Based Theorem

3. Domain Adaptation Algorithms
   - Ben-David et al.'s Domain Divergence
   - A First PAC-Bayesian Algorithm
   - A Second PAC-Bayesian Algorithm
   - A Neural Network / Representation Learning Algorithm

4. Conclusion and future works
**A New Change of Measure**

**Kullback-Leibler Change of Measure Inequality**

For any $P$ and $Q$ on $\mathcal{H}$, and for any $\phi : \mathcal{H} \rightarrow \mathbb{R}$, we have

$$
\mathbf{E}_{h \sim Q} \phi(h) \leq \text{KL}(Q\|P) + \ln \left( \mathbf{E}_{h \sim P} e^{\phi(h)} \right).
$$

**Rényi Change of Measure Inequality**

For any $P$ and $Q$ on $\mathcal{H}$, any $\phi : \mathcal{H} \rightarrow \mathbb{R}$, and for any $\alpha > 1$, we have

$$
\frac{\alpha}{\alpha - 1} \ln \mathbf{E}_{h \sim Q} \phi(h) \leq D_\alpha(Q\|P) + \ln \left( \mathbf{E}_{h \sim P} \phi(h)^{\frac{\alpha}{\alpha - 1}} \right),
$$

with

$$
D_\alpha(Q\|P) \overset{\text{def}}{=} \frac{1}{\alpha - 1} \ln \left[ \mathbf{E}_{h \sim P} \left( \frac{Q(h)}{P(h)} \right)^\alpha \right].
$$
Rényi-Based General Theorem

General theorem

[...] for any $\alpha > 1$, with probability at least $1-\delta$ over the choice of $S \sim D^n$,

$$\forall Q \text{ on } \mathcal{H}: \quad \ln \Delta \left(\hat{R}_S(G_Q), R_D(G_Q)\right) \leq \frac{1}{\alpha'} \left[ D_\alpha(Q\|P) + \ln \frac{\mathcal{I}^R_\Delta(n, \alpha')}{\delta} \right],$$

with

$$\mathcal{I}^R_\Delta(n, \alpha') \overset{\text{def}}{=} \sup_{r \in [0,1]} \left[ \sum_{k=0}^{n} \text{Bin}(k; n, r) \Delta(k/n, r)^{\alpha'} \right],$$

and $\alpha' := \frac{\alpha}{\alpha-1} > 1$. 

General theorem (Rényi-Based)

\[
\Pr_{S \sim D^n} \left( \forall Q \text{ on } \mathcal{H} : \ln \Delta \left( \hat{R}_S(G_Q), R_D(G_Q) \right) \leq \frac{1}{\alpha'} \left[ D_\alpha(Q\|P) + \ln \frac{I^R_{\Delta}(n, \alpha')}{\delta} \right] \right) \geq 1 - \delta.
\]

Proof.

\[
\alpha' := \frac{\alpha}{\alpha - 1}
\]

\[
\alpha' \cdot \ln \Delta \left( \mathbb{E}_{h \sim Q} \hat{R}_S(h), \mathbb{E}_{h \sim Q} R_D(h) \right)
\leq \alpha' \cdot \ln \mathbb{E}_{h \sim Q} \Delta \left( \hat{R}_S(h), R_D(h) \right)
\leq D_\alpha(Q\|P) + \ln \mathbb{E}_{h \sim P} \Delta \left( \hat{R}_S(h), R_D(h) \right)^{\alpha'}
\leq 1 - \delta
\]

Markov's Inequality

\[
\leq D_\alpha(Q\|P) + \ln \frac{1}{\delta} \sum_{h \sim P} \mathbb{E}_{h \sim P} \Delta \left( \hat{R}_S'(h), R_D(h) \right)^{\alpha'}
\]

Expectation swap

\[
= D_\alpha(Q\|P) + \ln \frac{1}{\delta} \sum_{h \sim P} \sum_{k=0}^n \text{Bin}(k; n, R_D(h)) \Delta \left( \frac{k}{n}, R_D(h) \right)^{\alpha'}
\]

Binomial law

\[
\leq D_\alpha(Q\|P) + \ln \frac{1}{\delta} \sup_{r \in [0, 1]} \left[ \sum_{k=0}^n \text{Bin}(k; n, r) \Delta \left( \frac{k}{n}, r \right)^{\alpha'} \right]
\]

Supremum over risk

\[
= D_\alpha(Q\|P) + \ln \frac{1}{\delta} I^R_{\Delta}(n, \alpha').
\]
Empirical Study
Majority votes of 500 decision trees on *Mushroom* dataset

\[ R_D(G_Q), \text{ Jensen's inequality, Change of measure, Markov's inequality, Supremum over risk} \]

Weak Decision Trees

Strong Decision Trees

\[ \times \text{KL}(Q\|P) \text{ and } \Delta := 2(q-p)^2 \]

\[ \times \text{KL}(Q\|P) \text{ and } \Delta := \text{kl}(q, p) \]

\[ \times \text{KL}(Q\|P) \text{ and } \Delta := 2(q-p)^2 \]

\[ \times \text{KL}(Q\|P) \text{ and } \Delta := \text{kl}(q, p) \]
Plan

1. Basic Definitions

2. PAC-Bayesian Theory
   - Majority Vote Classifiers
   - A Classical PAC-Bayesian Theorem
   - A General PAC-Bayesian Theorem
   - Transductive Learning
   - Rényi-Based Theorem

3. Domain Adaptation Algorithms
   - Ben-David et al.’s Domain Divergence
   - A First PAC-Bayesian Algorithm
   - A Second PAC-Bayesian Algorithm
   - A Neural Network / Representation Learning Algorithm

4. Conclusion and future works
Domain Adaptation

Assumption

Source and target examples are generated by different distributions

\[
S = \{ (x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n) \} \sim (D_S)^n
\]

\[
T = \{ (x_1, \cdot), (x_2, \cdot), \ldots, (x_n, \cdot) \} \sim (D_T)^n
\]
Our Domain Adaptation Setting

Binary classification tasks
- Input space: $\mathbb{R}^d$
- Labels: $\{-1, 1\}$

Two different data distributions
- Source domain: $D_S$
- Target domain: $D_T$

A domain adaptation learning algorithm is provided with

- A labeled source sample $S = \{(x^s_i, y^s_i)\}_{i=1}^n \sim (D_S)^n$,
- An unlabeled target sample $T = \{x^t_i\}_{i=1}^n \sim (D_T)^n$.

The goal is to build a classifier $h : \mathbb{R}^d \rightarrow \{-1, 1\}$ with a low target risk

$$R_{DT}(h) \overset{\text{def}}{=} \Pr_{(x^t, y^t) \sim D_T} [h(x^t) \neq y^t].$$
1 Basic Definitions

2 PAC-Bayesian Theory
   - Majority Vote Classifiers
   - A Classical PAC-Bayesian Theorem
   - A General PAC-Bayesian Theorem
   - Transductive Learning
   - Rényi-Based Theorem

3 Domain Adaptation Algorithms
   - Ben-David et al.’s Domain Divergence
   - A First PAC-Bayesian Algorithm
   - A Second PAC-Bayesian Algorithm
   - A Neural Network / Representation Learning Algorithm

4 Conclusion and future works
Divergence between source and target domains

Definition (Ben David et al., 2006)

Given two domain distributions $D_S$ and $D_T$, and a hypothesis class $\mathcal{H}$, the $\mathcal{H}$-divergence between $D_S$ and $D_T$ is

$$d_{\mathcal{H}}(D_S, D_T) \overset{\text{def}}{=} 2 \sup_{h \in \mathcal{H}} \left| \Pr_{x^s \sim D_S} [h(x^s) = 1] + \Pr_{x^t \sim D_T} [h(x^t) = -1] - 1 \right|.$$ 

The $\mathcal{H}$-divergence measures the ability of an hypothesis class $\mathcal{H}$ to discriminate between source $D_S$ and target $D_T$ distributions.
Bound on the target risk

**Theorem (Ben David et al., 2006)**

Let $\mathcal{H}$ be a hypothesis class of VC-dimension $d$. With probability $1 - \delta$ over the choice of samples $S \sim (D_S)^n$ and $T \sim (D_T)^n$, for every $h \in \mathcal{H}$:

$$R_{DT}(h) \leq \hat{R}_S(h) + \frac{4}{n} \sqrt{d \log \frac{2e n}{d}} + \log \frac{4}{\delta} + \hat{d}_\mathcal{H}(S, T) + \frac{4}{n^2} \sqrt{d \log \frac{2n}{d}} + \log \frac{4}{\delta} + \beta$$

with $\beta \geq \inf_{h^* \in \mathcal{H}} [R_{DS}(h^*) + R_{DT}(h^*)]$.

**Empirical risk on the source sample:**

$$\hat{R}_S(h) \overset{\text{def}}{=} \frac{1}{n} \sum_{i=1}^{n} I[h(x_s^i) \neq y_s^i].$$

**Empirical $\mathcal{H}$-divergence:**

$$\hat{d}_\mathcal{H}(S, T) \overset{\text{def}}{=} 2 \max_{h \in \mathcal{H}} \left[ \frac{1}{n} \sum_{i=1}^{n} I[h(x_s^i) = 1] + \frac{1}{n} \sum_{i=1}^{n} I[h(x_t^i) = -1] - 1 \right].$$
Plan

1. Basic Definitions

2. PAC-Bayesian Theory
   - Majority Vote Classifiers
   - A Classical PAC-Bayesian Theorem
   - A General PAC-Bayesian Theorem
   - Transductive Learning
   - Rényi-Based Theorem

3. Domain Adaptation Algorithms
   - Ben-David et al.’s Domain Divergence
   - A First PAC-Bayesian Algorithm
   - A Second PAC-Bayesian Algorithm
   - A Neural Network / Representation Learning Algorithm

4. Conclusion and future works
Nouvelle borne pour l’adaptation de domaine

\( \mathcal{H} \Delta \mathcal{H} \)-distance (Ben-David et al., 2006, 2010)

\[
d_{\mathcal{H} \Delta \mathcal{H}}(D_S, D_T) \overset{\text{def}}{=} 2 \sup_{h, h' \in \mathcal{H}} \left| \mathbb{E} \mathbb{I}[h(x^S) \neq h'(x^S)] - \mathbb{E} \mathbb{I}[h(x^T) \neq h'(x^T)] \right|
\]

Distributions disagreement

\[
dis_Q(D_S, D_T) \overset{\text{def}}{=} \left| d^D_T - d^D_S \right|
\]

Theorem

[...] with probability \( 1 - \delta \) over the choice of \( S \times T \sim (D_S \times D_T)^n \), we have

\[
\forall Q \text{ on } \mathcal{H} : \\
R_{D_T}(G_Q) \leq c' \hat{R}_S(G_Q) + a' \hat{dis}_Q(S, T) + \left( \frac{c'}{c} + \frac{2a'}{a} \right) \frac{\text{KL}(Q \parallel P) + \ln \frac{3}{\delta}}{n} + \lambda^*_Q + a' - 1
\]

where \( a' \overset{\text{def}}{=} \frac{2a}{1 - e^{-2a}} \) et \( c' \overset{\text{def}}{=} \frac{c}{1 - e^{-c}} \).
A New Domain Adaptation Algorithm
For linear classifiers

PAC-Bayes specialization to linear classifier (Langford and Shawe-Taylor, 2002)

- Linear classifier: \( h_w(x) = \text{sgn}[w \cdot x] \)
- Voters: \( \mathcal{H} = \{ h_v \mid v \in \mathbb{R}^d \} \)
- Prior \( P_0 \): isotropic Gaussian centered on \( 0 \)
- Posterior \( Q_w \): isotropic Gaussian centered on \( w \)

PBDA
Minimize:

\[
C \sum_{i=1}^{n} \Phi_c \left( y_i^S \frac{w \cdot x_i^S}{\|x_i^S\|} \right) + A \left| \sum_{i=1}^{n} \Phi_d \left( \frac{w \cdot x_i^S}{\|x_i^S\|} \right) - \Phi_d \left( \frac{w \cdot x_i^T}{\|x_i^T\|} \right) \right| + \frac{\|w\|^2}{2}
\]
Amazon Reviews Dataset

**Input**: product review (bag of words) — **Output**: positive or negative rating.

![Diagram of categories: Books, DVDs, Electronics, Kitchen](image)

<table>
<thead>
<tr>
<th>source → target</th>
<th>PBGD</th>
<th>SVM</th>
<th>DASVM</th>
<th>CODA</th>
<th>PBDA</th>
</tr>
</thead>
<tbody>
<tr>
<td>books→dvds</td>
<td>0.174</td>
<td>0.179</td>
<td>0.193</td>
<td>0.181</td>
<td>0.183</td>
</tr>
<tr>
<td>books→electronics</td>
<td>0.275</td>
<td>0.290</td>
<td>0.226</td>
<td>0.232</td>
<td>0.263</td>
</tr>
<tr>
<td>books→kitchen</td>
<td>0.236</td>
<td>0.251</td>
<td>0.179</td>
<td>0.215</td>
<td>0.229</td>
</tr>
<tr>
<td>dvds→books</td>
<td>0.192</td>
<td>0.203</td>
<td>0.202</td>
<td>0.217</td>
<td>0.197</td>
</tr>
<tr>
<td>dvds→electronics</td>
<td>0.256</td>
<td>0.269</td>
<td>0.186</td>
<td>0.214</td>
<td>0.241</td>
</tr>
<tr>
<td>dvds→kitchen</td>
<td>0.211</td>
<td>0.232</td>
<td>0.183</td>
<td>0.181</td>
<td>0.186</td>
</tr>
<tr>
<td>electronics→books</td>
<td>0.268</td>
<td>0.287</td>
<td>0.305</td>
<td>0.275</td>
<td>0.232</td>
</tr>
<tr>
<td>electronics→dvds</td>
<td>0.245</td>
<td>0.267</td>
<td>0.214</td>
<td>0.239</td>
<td>0.221</td>
</tr>
<tr>
<td>electronics→kitchen</td>
<td>0.127</td>
<td>0.129</td>
<td>0.149</td>
<td>0.134</td>
<td>0.141</td>
</tr>
<tr>
<td>kitchen→books</td>
<td>0.255</td>
<td>0.267</td>
<td>0.259</td>
<td>0.247</td>
<td>0.247</td>
</tr>
<tr>
<td>kitchen→dvds</td>
<td>0.244</td>
<td>0.253</td>
<td>0.198</td>
<td>0.238</td>
<td>0.233</td>
</tr>
<tr>
<td>kitchen→electronics</td>
<td>0.235</td>
<td>0.149</td>
<td>0.157</td>
<td>0.153</td>
<td>0.129</td>
</tr>
<tr>
<td><strong>Mean</strong></td>
<td>0.226</td>
<td>0.231</td>
<td><strong>0.204</strong></td>
<td>0.210</td>
<td>0.208</td>
</tr>
</tbody>
</table>
Plan

1. Basic Definitions

2. PAC-Bayesian Theory
   - Majority Vote Classifiers
   - A Classical PAC-Bayesian Theorem
   - A General PAC-Bayesian Theorem
   - Transductive Learning
   - Rényi-Based Theorem

3. Domain Adaptation Algorithms
   - Ben-David et al.’s Domain Divergence
   - A First PAC-Bayesian Algorithm
   - A Second PAC-Bayesian Algorithm
   - A Neural Network / Representation Learning Algorithm

4. Conclusion and future works
A New Perspective on Domain Adaptation

**Theorem**

For all pairs $D_S$ and $D_T$ on $\mathcal{X} \times \mathcal{Y}$, for all set $\mathcal{H}$ of voters, and for all $q > 0$,

$$\forall Q \text{ sur } \mathcal{H}, \quad R_{DT}(G_Q) \leq \frac{1}{2} d^D_Q + \beta_q(D_T \| D_S) \times \left[ e^D_Q \right]^{1 - \frac{1}{q}}.$$

where $\beta_q(D_T \| D_S) = \left[ \mathbb{E}_{(x,y) \sim D_S} \left( \frac{D_T(x, y)}{D_S(x, y)} \right)^q \right]^{\frac{1}{q}}$,

and $e^D_Q \overset{\text{def}}{=} \mathbb{E}_{(x,y) \sim D} \mathbb{E}_{h_1 \sim Q} \mathbb{E}_{h_2 \sim Q} \mathbb{I}[h_1(x) \neq y] \times \mathbb{I}[h_2(x) \neq y]$.

**DALC**

Minimize:

$$A \sum_{i=1}^{n_t} \Phi_d \left( \frac{w \cdot x_i^T}{\|x_i\|} \right) + B \sum_{i=1}^{n_s} \Phi_{err} \left( \frac{y_i^S w \cdot x_i^S}{\|x_i^S\|} \right) + \frac{\|w\|^2}{2}$$
### Amazon Reviews Dataset

**Input**: product review (bag of words) — **Output**: positive or negative rating.

![Diagram](image)

<table>
<thead>
<tr>
<th>source → target</th>
<th>PBGD</th>
<th>SVM</th>
<th>DASVM</th>
<th>CODA</th>
<th>PBDA</th>
<th>DALC</th>
</tr>
</thead>
<tbody>
<tr>
<td>books→dvds</td>
<td>0.174</td>
<td>0.179</td>
<td>0.193</td>
<td>0.181</td>
<td>0.183</td>
<td>0.178</td>
</tr>
<tr>
<td>books→electronics</td>
<td>0.275</td>
<td>0.290</td>
<td>0.226</td>
<td>0.232</td>
<td>0.263</td>
<td>0.212</td>
</tr>
<tr>
<td>books→kitchen</td>
<td>0.236</td>
<td>0.251</td>
<td><strong>0.179</strong></td>
<td>0.215</td>
<td>0.229</td>
<td>0.194</td>
</tr>
<tr>
<td>dvds→books</td>
<td>0.192</td>
<td>0.203</td>
<td>0.202</td>
<td>0.217</td>
<td>0.197</td>
<td><strong>0.186</strong></td>
</tr>
<tr>
<td>dvds→electronics</td>
<td>0.256</td>
<td>0.269</td>
<td><strong>0.186</strong></td>
<td>0.214</td>
<td>0.241</td>
<td>0.245</td>
</tr>
<tr>
<td>dvds→kitchen</td>
<td>0.211</td>
<td>0.232</td>
<td>0.183</td>
<td>0.181</td>
<td>0.186</td>
<td><strong>0.175</strong></td>
</tr>
<tr>
<td>electronics→books</td>
<td>0.268</td>
<td>0.287</td>
<td>0.305</td>
<td>0.275</td>
<td><strong>0.232</strong></td>
<td>0.240</td>
</tr>
<tr>
<td>electronics→dvds</td>
<td>0.245</td>
<td>0.267</td>
<td><strong>0.214</strong></td>
<td>0.239</td>
<td>0.221</td>
<td>0.256</td>
</tr>
<tr>
<td>electronics→kitchen</td>
<td>0.127</td>
<td>0.129</td>
<td>0.149</td>
<td>0.134</td>
<td>0.141</td>
<td><strong>0.123</strong></td>
</tr>
<tr>
<td>kitchen→books</td>
<td>0.255</td>
<td>0.267</td>
<td>0.259</td>
<td>0.247</td>
<td>0.247</td>
<td><strong>0.236</strong></td>
</tr>
<tr>
<td>kitchen→dvds</td>
<td>0.244</td>
<td>0.253</td>
<td><strong>0.198</strong></td>
<td>0.238</td>
<td>0.233</td>
<td>0.225</td>
</tr>
<tr>
<td>kitchen→electronics</td>
<td>0.235</td>
<td>0.149</td>
<td>0.157</td>
<td>0.153</td>
<td><strong>0.129</strong></td>
<td>0.131</td>
</tr>
<tr>
<td><strong>Mean</strong></td>
<td>0.226</td>
<td>0.231</td>
<td>0.204</td>
<td>0.210</td>
<td>0.208</td>
<td><strong>0.200</strong></td>
</tr>
</tbody>
</table>
Plan

1 Basic Definitions

2 PAC-Bayesian Theory
   - Majority Vote Classifiers
   - A Classical PAC-Bayesian Theorem
   - A General PAC-Bayesian Theorem
   - Transductive Learning
   - Rényi-Based Theorem

3 Domain Adaptation Algorithms
   - Ben-David et al.'s Domain Divergence
   - A First PAC-Bayesian Algorithm
   - A Second PAC-Bayesian Algorithm
   - A Neural Network / Representation Learning Algorithm

4 Conclusion and future works
Bound on the target risk

**Theorem (Ben David et al., 2006)**

Let $\mathcal{H}$ be a hypothesis class of VC-dimension $d$. With probability $1 - \delta$ over the choice of samples $S \sim (D_S)^n$ and $T \sim (D_T)^n$, for every $h \in \mathcal{H}$:

$$R_{DT}(h) \leq \hat{R}_S(h) + \frac{4}{n} \sqrt{d \log \frac{2en}{d} + \log \frac{4}{\delta}} + \hat{d}_\mathcal{H}(S, T) + \frac{4}{n^2} \sqrt{d \log \frac{2n}{d} + \log \frac{4}{\delta}} + \beta$$

with $\beta \geq \inf_{h^* \in \mathcal{H}} [R_{DS}(h^*) + R_{DT}(h^*)]$.

**Target risk** $R_{DT}(h)$ is low if, given $S$ and $T$, $\hat{R}_S(h)$ is small, i.e., $h \in \mathcal{H}$ is good on and $\hat{d}_\mathcal{H}(S, T)$ is small, i.e., all $h' \in \mathcal{H}$ are bad on
Empirical $\mathcal{H}$-divergence

$$
\hat{d}_\mathcal{H}(S, T) \overset{\text{def}}{=} 2 \max_{h \in \mathcal{H}} \left[ \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}[h(x^s_i) = 1] + \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}[h(x^t_i) = -1] - 1 \right].
$$

We estimate the $\mathcal{H}$-divergence by a logistic regressor that models the probability that a given input (either $x^s$ or $x^t$) is from the source domain:

$$
o(h(x)) \overset{\text{def}}{=} \text{sigm}(d + w^\top h(x)).
$$

Given a representation output by the hidden layer $h(\cdot)$:

$$
\hat{d}_\mathcal{H}(h(S), h(T)) \approx 2 \max_{w, d} \left[ \frac{1}{n} \sum_{i=1}^{n} \log(o(h(x^s_i))) + \frac{1}{n} \sum_{i=1}^{n} \log(1 - o(h(x^t_i))) - 1 \right].
$$
Domain-Adversarial Neural Network (DANN)

\[
\min_{W,V,b,c} \left[ \frac{1}{n} \sum_{i=1}^{n} - \log (f_{y_i}^S(x_i^S)) + \lambda \max_{w,d} \left( \frac{1}{n} \sum_{i=1}^{n} \log (o(h(x_i^S))) + \frac{1}{n} \sum_{i=1}^{n} \log (1 - o(h(x_t^S))) \right) \right],
\]

where \( \lambda > 0 \) weights the domain adaptation regularization term.

Given a source sample \( S = \{(x_i^S, y_i^S)\}_{i=1}^{m} \sim (D_S)^m \), and a target sample \( T = \{(x_i^T)\}_{i=1}^{m} \sim (D_T)^m \),

1. Pick a \( x^s \in S \) and \( x^t \in T \)
2. Update \( V \) towards \( f(h(x^s)) = y^s \)
3. Update \( W \) towards \( f(h(x^s)) = y^s \)
4. Update \( w \) towards \( o(h(x^s)) = 1 \) and \( o(h(x^t)) = -1 \)
5. Update \( W \) towards \( o(h(x^s)) = -1 \) and \( o(h(x^t)) = 1 \)

**DANN finds a representation \( h(\cdot) \) that are good on \( S \); but unable to discriminate between \( S \) and \( T \).**
Toy Dataset

**Standard Neural Network (NN)**

- Trained to classify source
- Trained to classify domains

**Domain-Adversarial Neural Networks (DANN)**

- Classification output: $f(h(x))$
- Domain output: $o(h(x))$
Amazon Reviews

**Input**: product review (bag of words) — **Output**: positive or negative rating.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>DANN</th>
<th>NN</th>
</tr>
</thead>
<tbody>
<tr>
<td>books → dvd</td>
<td>0.201</td>
<td>0.199</td>
</tr>
<tr>
<td>books → electronics</td>
<td>0.246</td>
<td>0.251</td>
</tr>
<tr>
<td>books → kitchen</td>
<td>0.230</td>
<td>0.235</td>
</tr>
<tr>
<td>dvd → books</td>
<td>0.247</td>
<td>0.261</td>
</tr>
<tr>
<td>dvd → electronics</td>
<td>0.247</td>
<td>0.256</td>
</tr>
<tr>
<td>dvd → kitchen</td>
<td>0.227</td>
<td>0.227</td>
</tr>
<tr>
<td>electronics → books</td>
<td>0.280</td>
<td>0.281</td>
</tr>
<tr>
<td>electronics → dvd</td>
<td>0.273</td>
<td>0.277</td>
</tr>
<tr>
<td>electronics → kitchen</td>
<td>0.148</td>
<td>0.149</td>
</tr>
<tr>
<td>kitchen → books</td>
<td>0.283</td>
<td>0.288</td>
</tr>
<tr>
<td>kitchen → dvd</td>
<td>0.261</td>
<td>0.261</td>
</tr>
<tr>
<td>kitchen → electronics</td>
<td>0.161</td>
<td>0.161</td>
</tr>
</tbody>
</table>
Deeper and deeper...

To appear in JMLR: **Domain-Adversarial Neural Networks.**
by Ganin, Ustinova, Ajakan, Germain, Larochelle, Laviolette, Marchand and Lempitsky

![Diagram of domain-adversarial neural networks.]
Plan

1. Basic Definitions

2. PAC-Bayesian Theory
   - Majority Vote Classifiers
   - A Classical PAC-Bayesian Theorem
   - A General PAC-Bayesian Theorem
   - Transductive Learning
   - Rényi-Based Theorem

3. Domain Adaptation Algorithms
   - Ben-David et al.'s Domain Divergence
   - A First PAC-Bayesian Algorithm
   - A Second PAC-Bayesian Algorithm
   - A Neural Network / Representation Learning Algorithm

4. Conclusion and future works
An original PAC-Bayesian approach

- General theorem from which we recover existing results;
- Modular proof, easy to adapt to various frameworks.

Domain adaptation algorithms

- Two algorithms for linear classifiers derived from PAC-Bayesian bounds;
- One *representation learning* Network inspired by the seminal work of Ben-David et al.
Perspectives

- Explore the relationships between PAC-Bayesian and truly Bayesian approaches;
- Speed-up domain adaptation algorithms with stochastic gradient;
- Go beyond simple binary classification setting;
- Apply PAC-Bayes to your problems!

*If you only have a hammer, you tend to see every problem as a nail.*

— Abraham Maslow, 1966