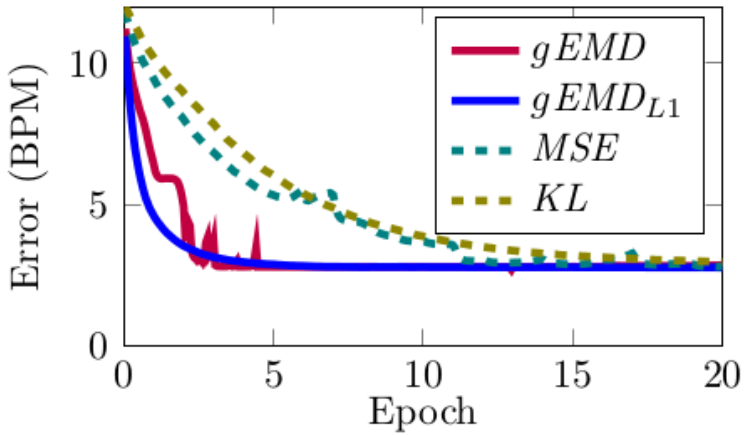
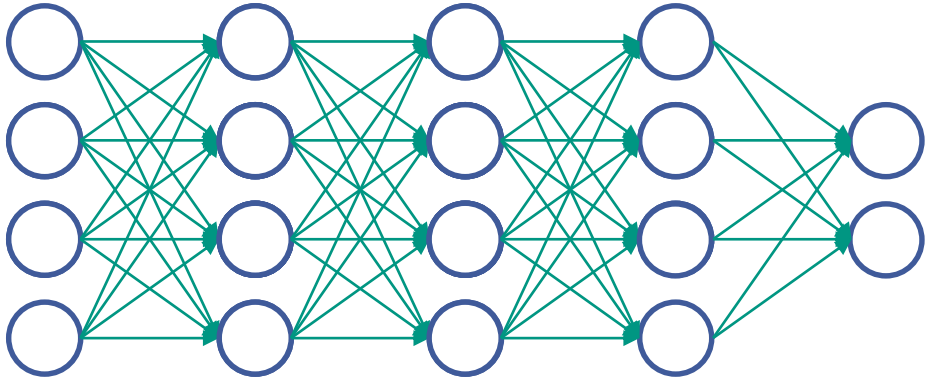
Karlsruhe Institute of Technology

Optimizing Problems with the Wasserstein Loss and the Importance of Choosing the Right Loss Function

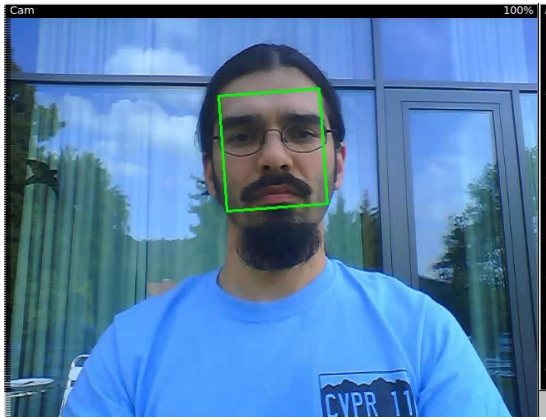
Manuel Martinez
manuel.martinez@kit.edu
Computer Vision and Human Interaction Lab
Prof. Rainer Stiefelhagen

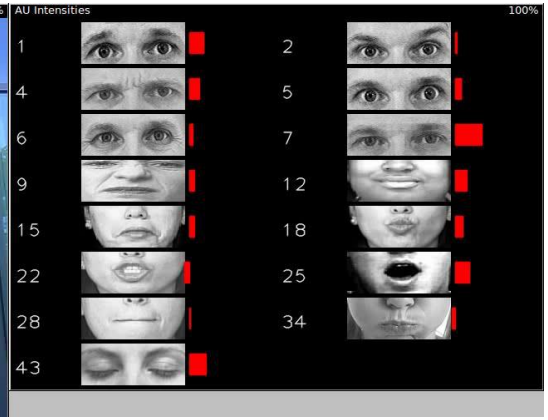
INSTITUTE FOR ANTHROPOMATICS, CV-HCI

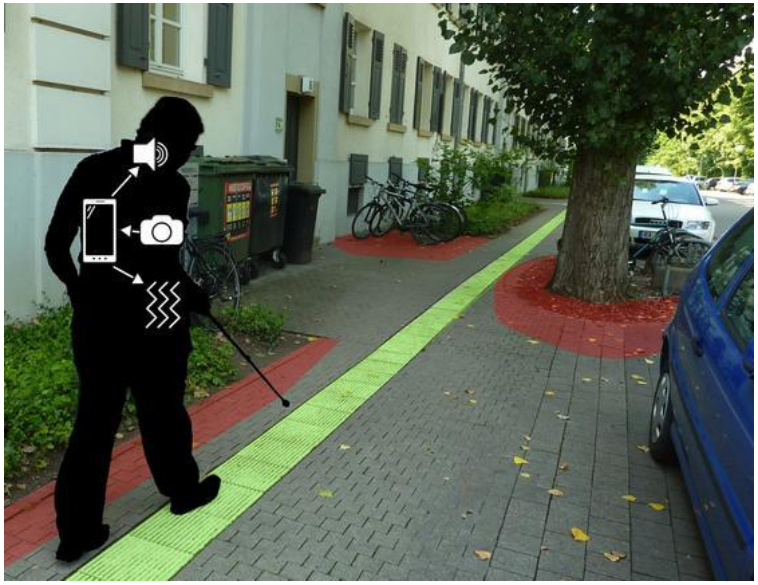




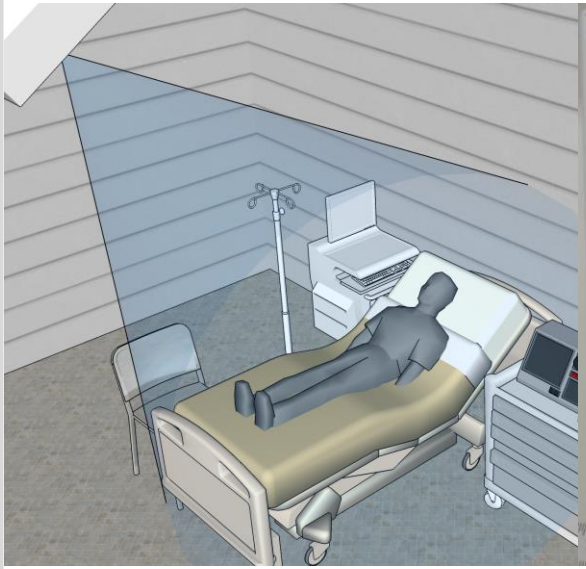
Computer Vision for Human Computer Interaction
Prof. Rainer Stiefelhagen

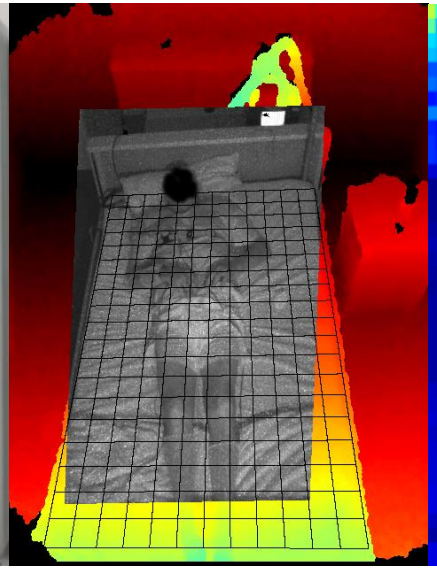


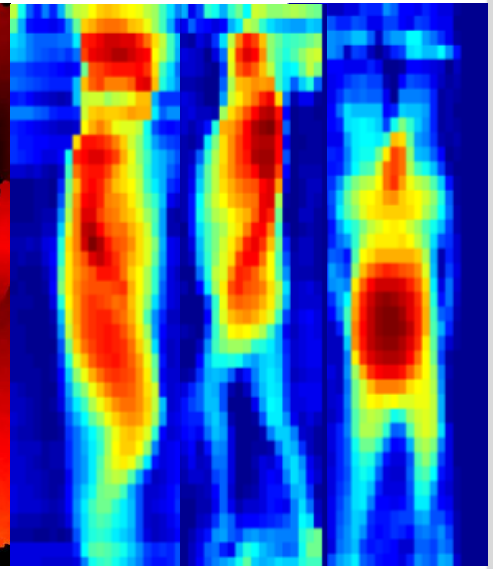


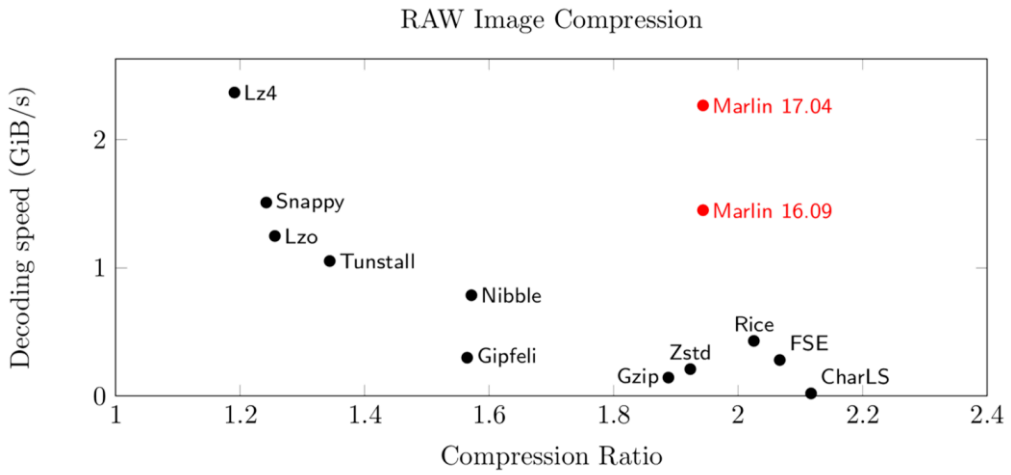


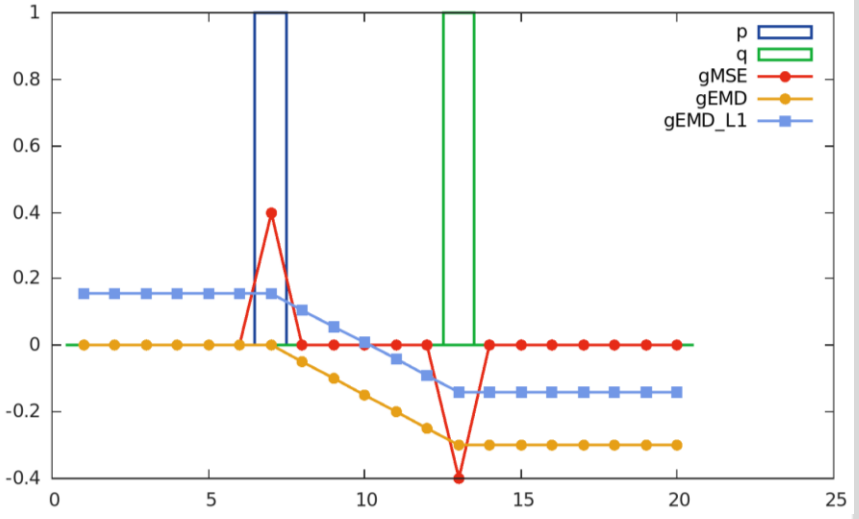
What do I do?







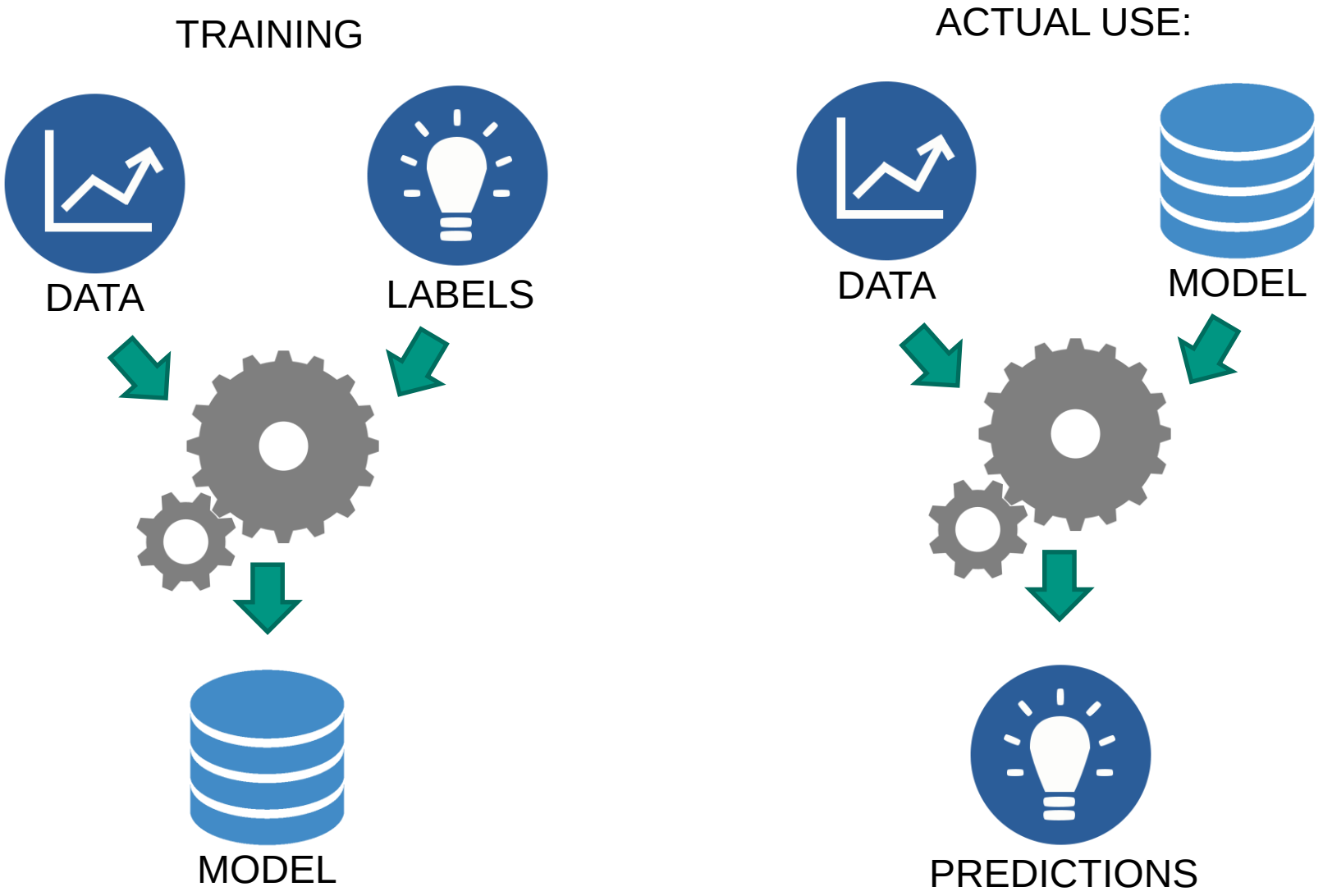

RAW Image Compression
Decoding speed (GiB/s)
Compression Ratio
Lz4
Snappy
Lzo
Tunstall
Nibble
Gipfeli
Gzip
Zstd
Rice
FSE
CharLS
Marlin 17.04
Marlin 16.09


p
q
gMSE
gEMD
gEMD_L1

Organization

- What is a Loss function (a.k.a. Criterion)
- The Wasserstein Distance
  - Mallow's Moments
  - Wasserstein Distance
  - Earth Mover's Distance
  - Not Wasserstein Distance
  - Sinkhorn Distance
- Using the Wasserstein Distance as a Loss
  - Sinkhorn Distance
  - Pure Wasserstein Distance
  - Squared Wasserstein Distance
- Examples
  - Breath Rate Prediction
  - ImageNet Experiments
  - Wasserstein GANs
  - Etc.

What is Machine Learning, Artificial Intelligence, or Pattern Recognition?



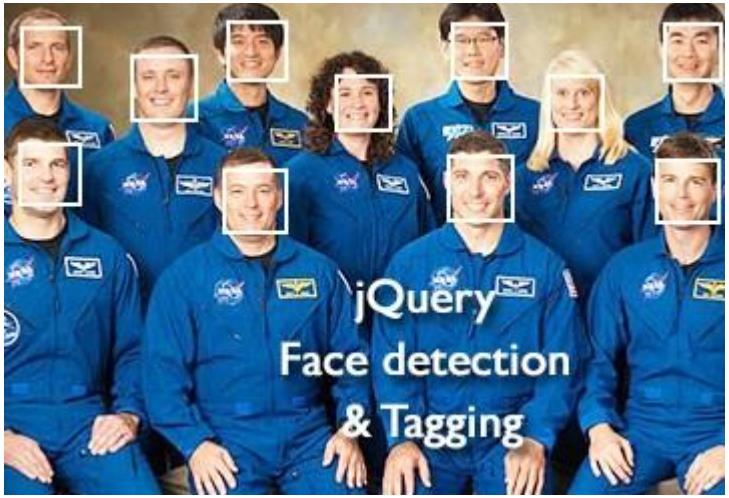
News & Analysis

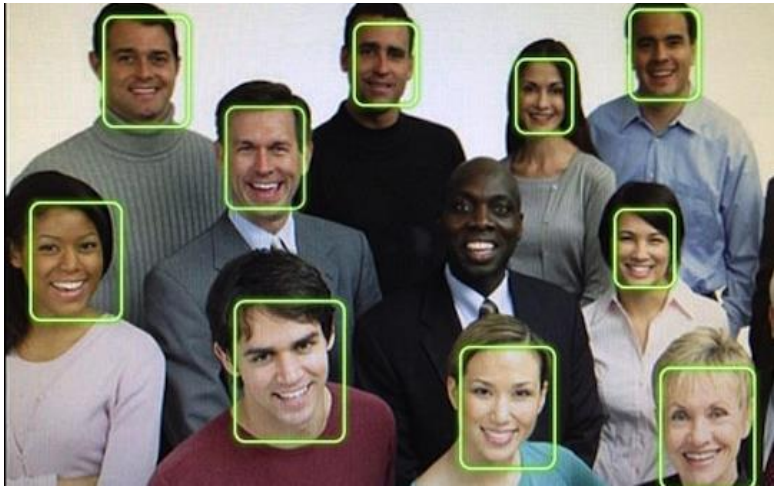
[redacted] Responds to Claim of 'Racist' Webcams

In a video posted to YouTube this week, two co-workers - one white and one black - tried out the webcam face-tracking software on an [redacted] computer. It is supposed to follow users as they move, but it fails to recognize Desi, a black man.

DATA MODEL

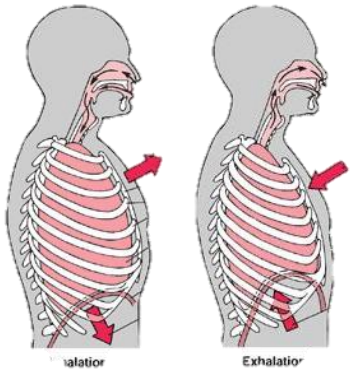
By Chloe Albanesius December 22, 2009 8:35AM EST

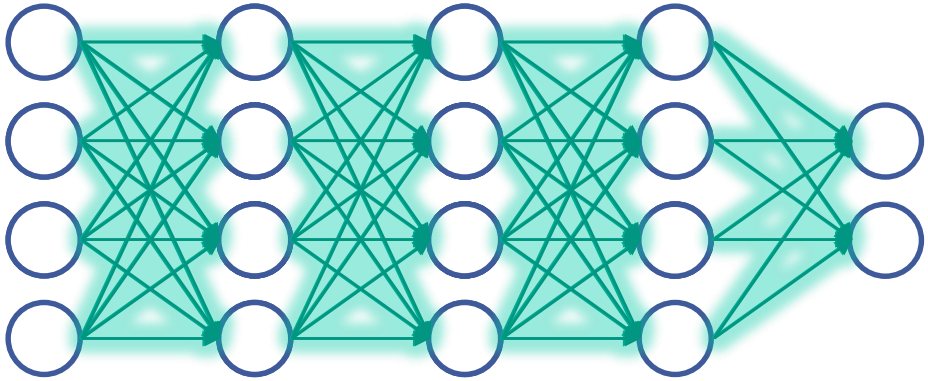


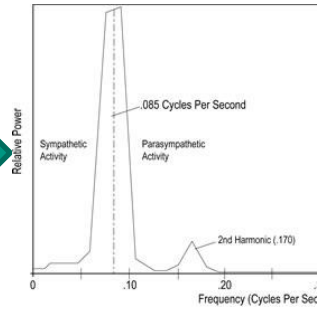


Example time!

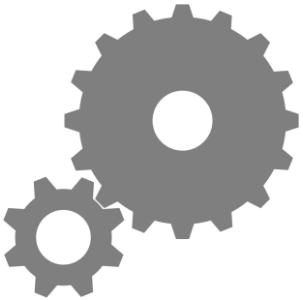
$$\mathbf{a}_{n+1} = \mathbf{a}_n - \gamma \nabla F(\mathbf{a}_n)$$







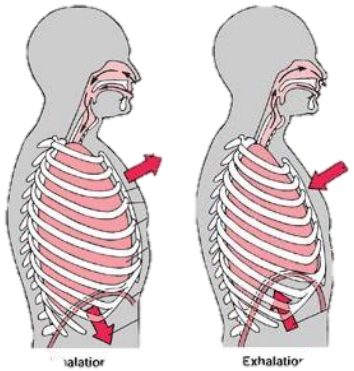
DATA

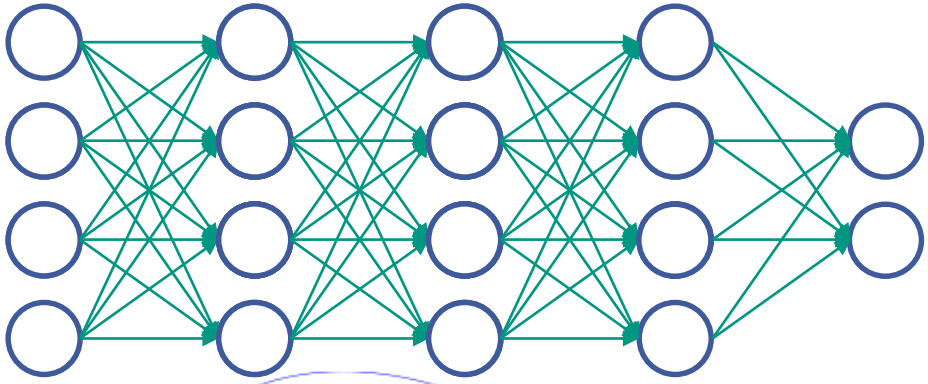


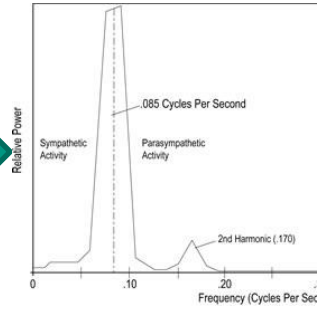
LABELS

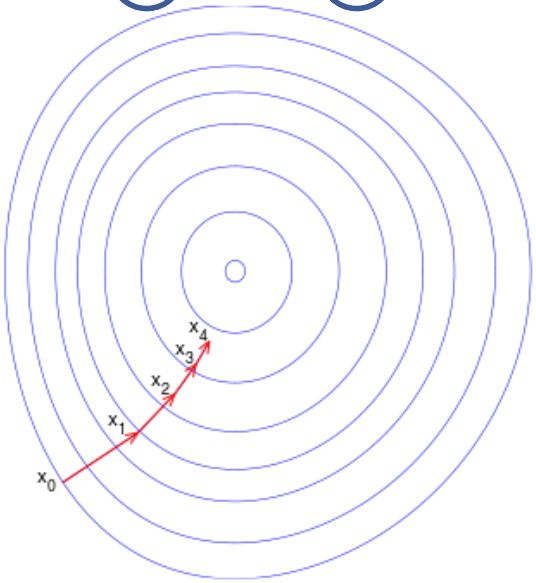
Example time!

$$\mathbf{a}_{n+1} = \mathbf{a}_n - \gamma \nabla F(\mathbf{a}_n)$$



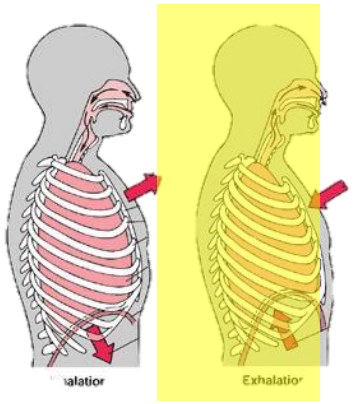


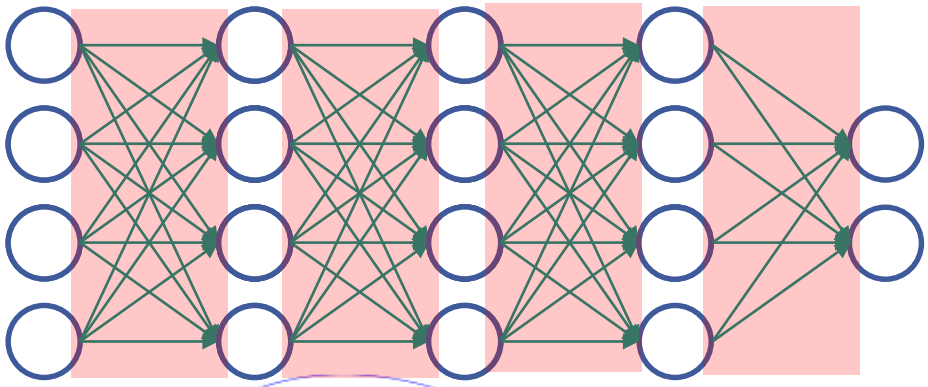


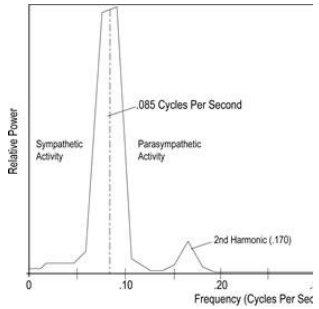


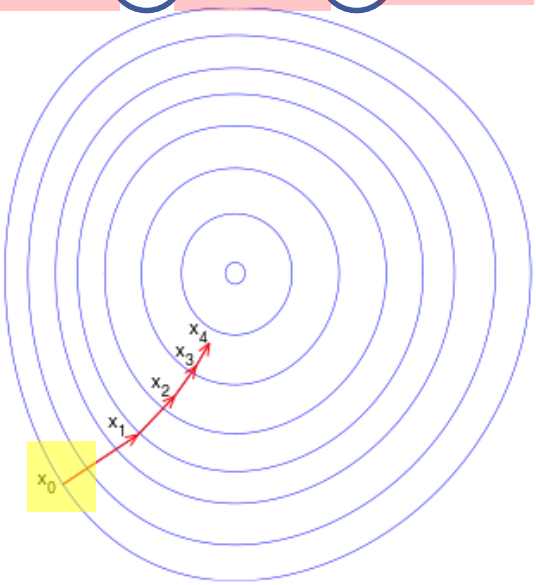
Example time!

$$\mathbf{a}_{n+1} = \mathbf{a}_n - \gamma \nabla F(\mathbf{a}_n)$$



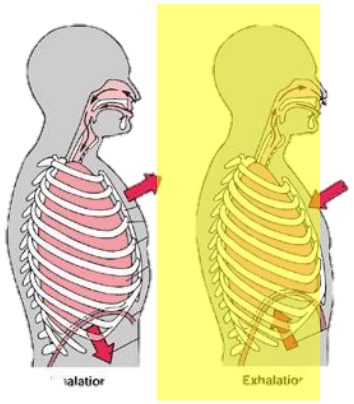


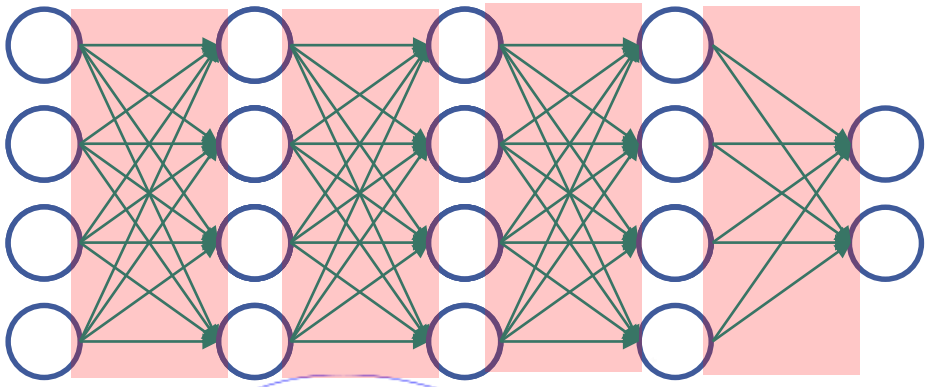


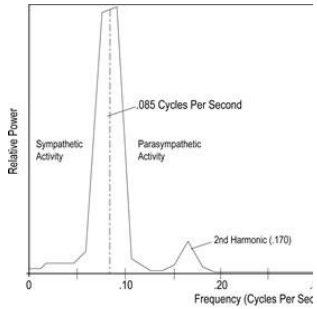


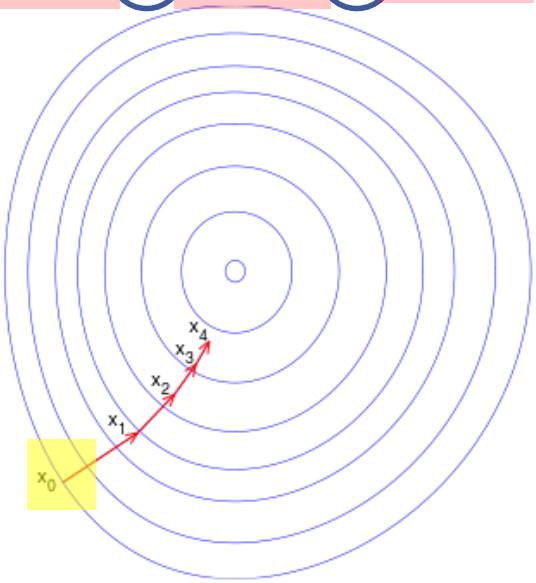
Example time!

$$\mathbf{a}_{n+1} = \mathbf{a}_n - \gamma \nabla F(\mathbf{a}_n)$$



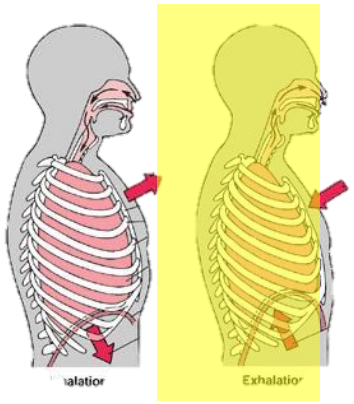


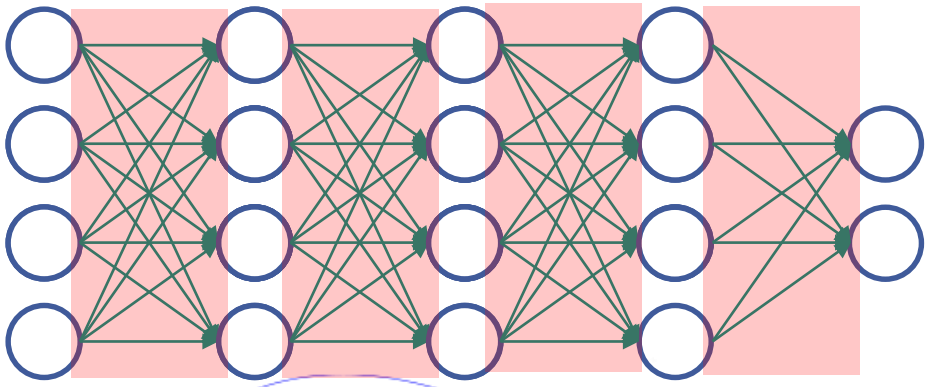


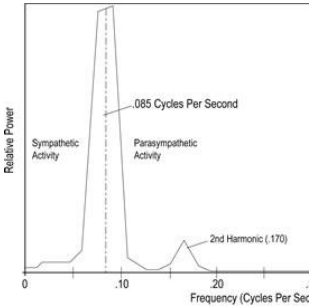


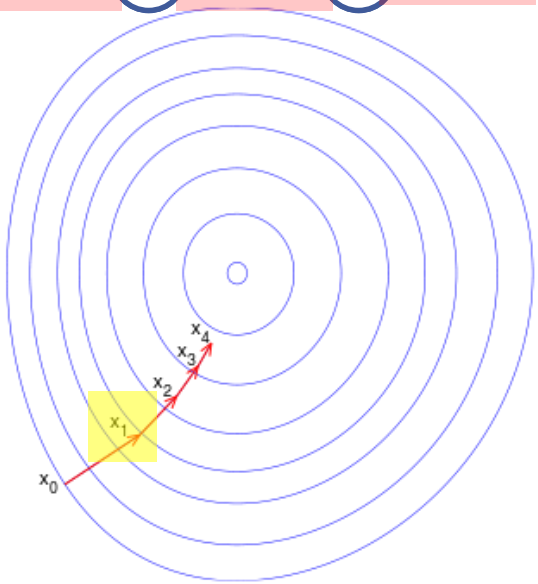
Example time!

$$\mathbf{a}_{n+1} = \mathbf{a}_n - \gamma \nabla F(\mathbf{a}_n)$$







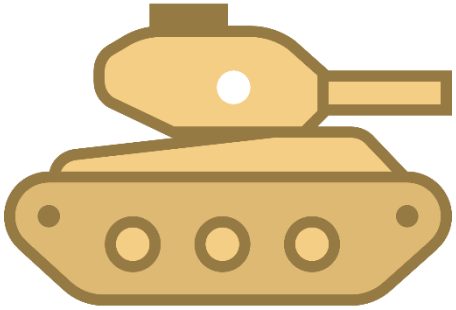


Analogy time!

$$\mathbf{a}_{n+1} = \mathbf{a}_n - \gamma \nabla F(\mathbf{a}_n)$$

Limited nº of uses!

What matters the most.



Analogy time!

$$\mathbf{a}_{n+1} = \mathbf{a}_n - \gamma \nabla F(\mathbf{a}_n)$$

$$pos_{n+1} = pos_n - \gamma \nabla_{pos_n} L(pos_n, target)$$

L: Loss Function
(a.k.a) Criterion

Limited nº of uses!

Manhattan Distance

The Manhattan distance is the L_1-norm of the difference, a special case of the Minkowski distance with p=1 and equivalent to the sum of absolute difference.

$$d_1 \equiv d_{\mathbf{SAD}} : (x, y) \mapsto \|x - y\|_1 = \sum_{i=1}^{n} |x_i - y_i|$$

https://numerics.mathdotnet.com/distance.html

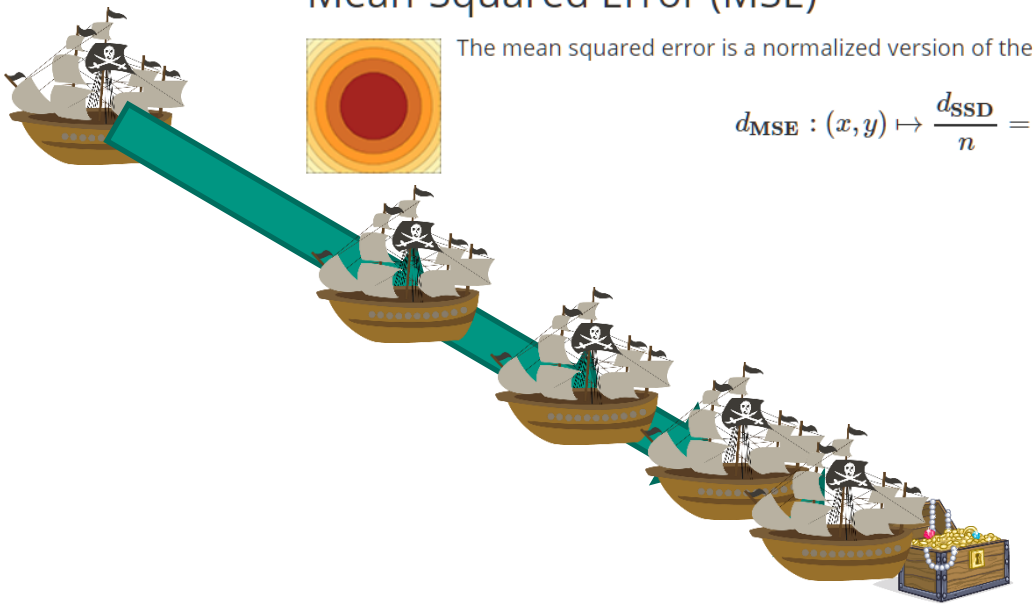
Euclidean Distance

The euclidean distance is the L_2-norm of the difference, a special case of the Minkowski distance with p=2. It is the natural distance in a geometric interpretation.

$$d_2 : (x, y) \mapsto \|x - y\|_2 = \sqrt{d_{\mathbf{SSD}}} = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

https://numerics.mathdotnet.com/distance.html

Mean-Squared Error (MSE)



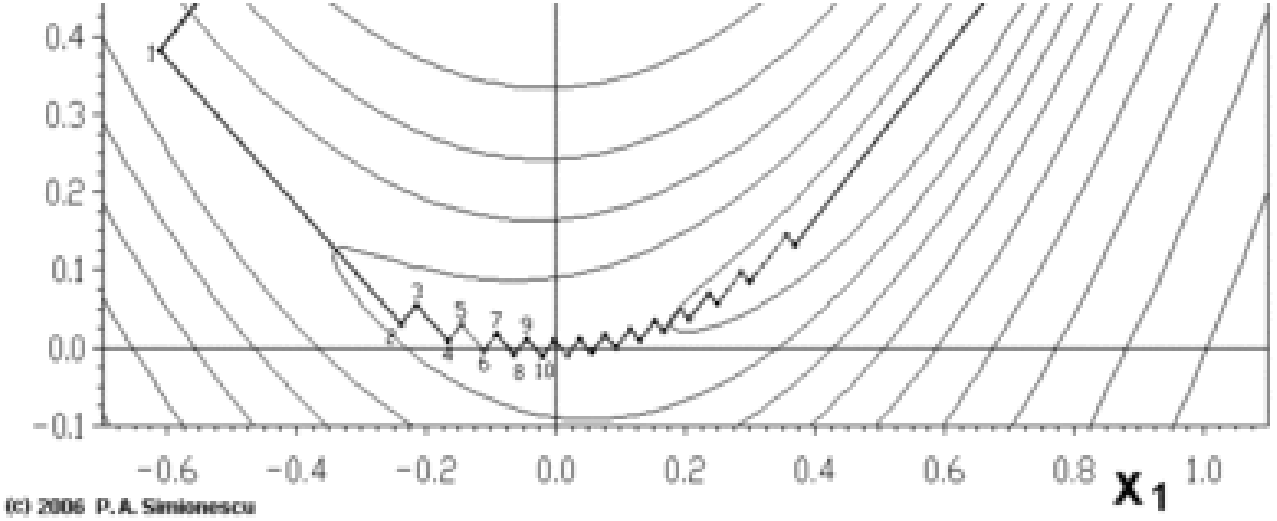
The mean squared error is a normalized version of the sum of squared difference.

$$d_{\mathbf{MSE}} : (x, y) \mapsto \frac{d_{\mathbf{SSD}}}{n} = \frac{\|x - y\|_2^2}{n} = \frac{1}{n} \sum_{i=1}^{n} (x_i - y_i)^2$$

https://numerics.mathdotnet.com/distance.html

Notes:

- Sense of Scale: Euclidean norms (or L1 norms in general) do not have sense of scale.
- On High Dimensional Problems we have the curse of dimensionality.
- MSE: Works well with convex problems with quadratic structure.
- Each Dimension should have similar converging properties.



How do we choose the right loss/criterion?

- Classification criterions:
  - `BCECriterion` : binary cross-entropy for `Sigmoid` (two-class version of `ClassNLLCriterion`);
  - `ClassNLLCriterion` : negative log-likelihood for `LogSoftMax` (multi-class);
  - `CrossEntropyCriterion` : combines `LogSoftMax` and `ClassNLLCriterion`;
  - `ClassSimplexCriterion` : A simplex embedding criterion for classification.
  - `MarginCriterion` : two class margin-based loss;
  - `SoftMarginCriterion` : two class softmargin-based loss;
  - `MultiMarginCriterion` : multi-class margin-based loss;
  - `MultiLabelMarginCriterion` : multi-class multi-classification margin-based loss;
  - `MultiLabelSoftMarginCriterion` : multi-class multi-classification loss based on binary cross-entropy;

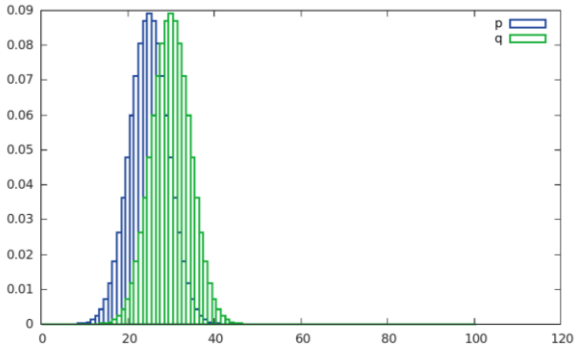
From Torch7

How do we choose the right loss/criterion?

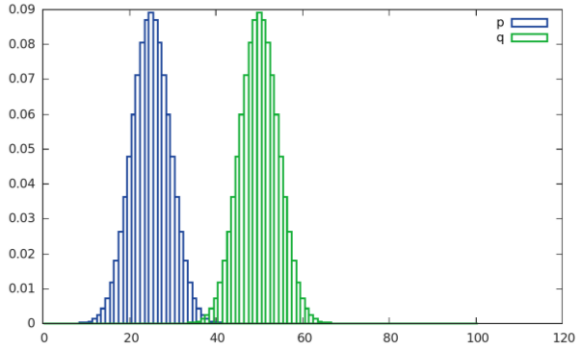
- Regression criterions:
  - `AbsCriterion` : measures the mean absolute value of the element-wise difference between input;
  - `SmoothL1Criterion` : a smooth version of the AbsCriterion;
  - `MSECriterion` : mean square error (a classic);
  - `SpatialAutoCropMSECriterion` : Spatial mean square error when the input is spatially smaller than the comparing their spatial overlap;
  - `DistKLDivCriterion` : Kullback–Leibler divergence (for fitting continuous probability distributions);
- Embedding criterions (measuring whether two inputs are similar or dissimilar):
  - `HingeEmbeddingCriterion` : takes a distance as input;
  - `L1HingeEmbeddingCriterion` : L1 distance between two inputs;
  - `CosineEmbeddingCriterion` : cosine distance between two inputs;
  - `DistanceRatioCriterion` : Probabilistic criterion for training siamese model with triplets.

From Torch7

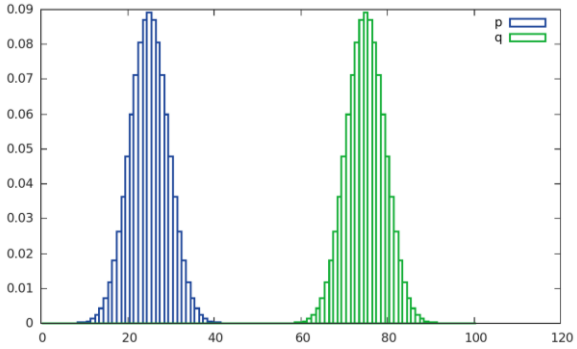
Which Loss would you choose for learning a 1D histogram?



MSE: 13.54
EMD: 5



MSE: 50.44
EMD: 25



MSE: 50.46
EMD: 50

- Computer Science: Earth Mover's Distance
- Math: Wasserstein Distance
- Statistics: Mallows Distance

The naming is not confusing at all #irony

Browse Conferences > Computer Vision, 2001. ICCV 2...

The Earth Mover's distance is the Mallows distance: some insights from statistics

Sign In or Purchase to View Full Text

78 Paper Citations | 4 Patent Citations | 607 Full Text Views

Related Articles

Technology challenges for building Internet-scale ubiquitous computing

Performance evaluation of a probabilistic replica selection algorithm

View All

2 Author(s) — E. Levina ; P. Bickel — View All Authors

Abstract | Authors | Figures | References | Citations | Keywords | Metrics | Media

Abstract:
The Earth Mover's distanc1e was first introduced as a purely empirical ways to measure texture and color similarities. We show that it has a rigorous probabilistic interpretation and is conceptually equivalent to the Mallows distance on probability distributions. The two distances are exactly the same when applied to probability distributions, but behave differently when applied to unnormalized distributions with different masses, called signatures. We discuss the advantages and disadvantages of both distances, and statistical issues involved in computing them from data. We also report some texture classification results for the Mallows distance applied to texture features and compare several ways of estimating feature distributions. In addition, we list some known probabilistic properties of this distance.

Published in: Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on

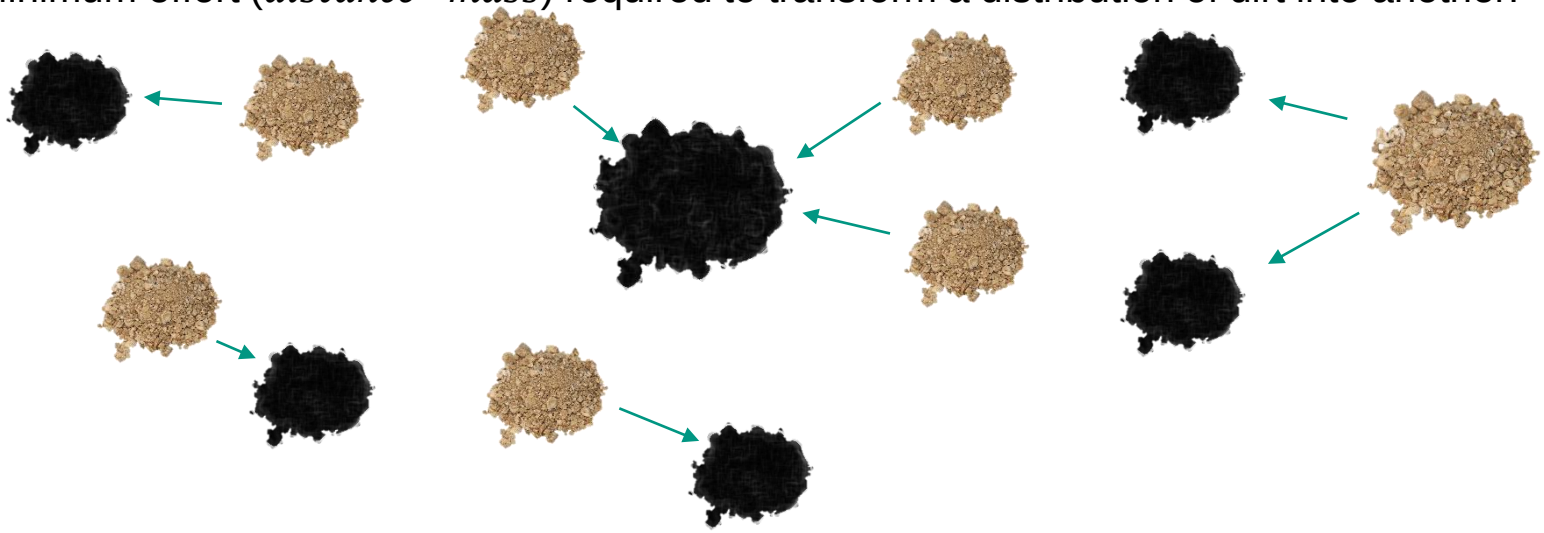
Earth Mover's Distance

2.2. **Optimal Transportation.** Given a $d \times d$ cost matrix M, the cost of mapping r to c using a transportation matrix (or joint probability) P can be quantified as $\langle P, M \rangle$. The following problem:

$$d_M(r, c) \stackrel{\text{def}}{=} \min_{P \in U(r,c)} \langle P, M \rangle.$$

is called an *optimal transportation* problem between r and c given cost M. An

- Minimum effort ($distance \cdot mass$) required to transform a distribution of dirt into another:



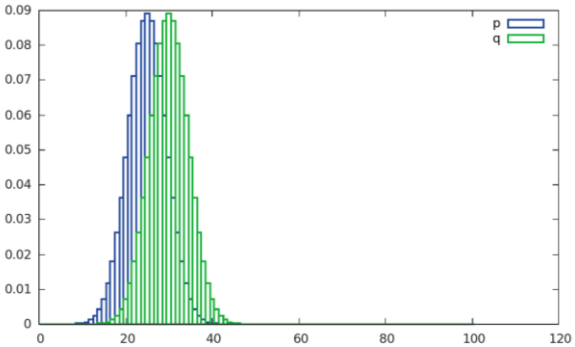
Can we use it in DL? No! It is very slow to calculate!

"Sinkhorn distances: Lightspeed computation of optimal transport", Cuturi, NIPS 2013

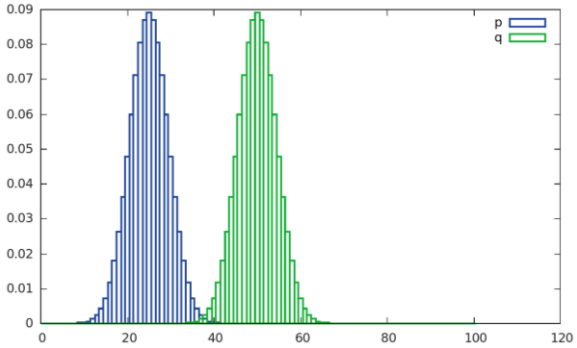
ALSO:
SPECIAL CASE FOR HISTOGRAMS

Earth Mover's Distance

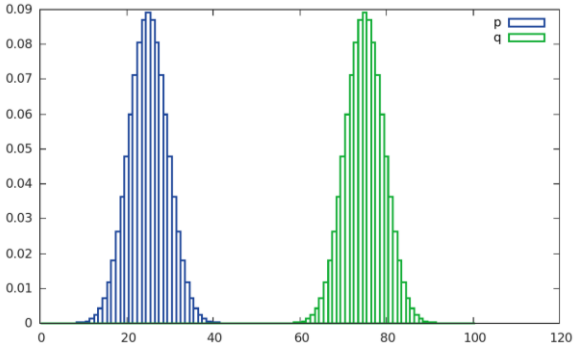
- Is the de-facto L1 distance between histograms.



EMD: 5



EMD: 25



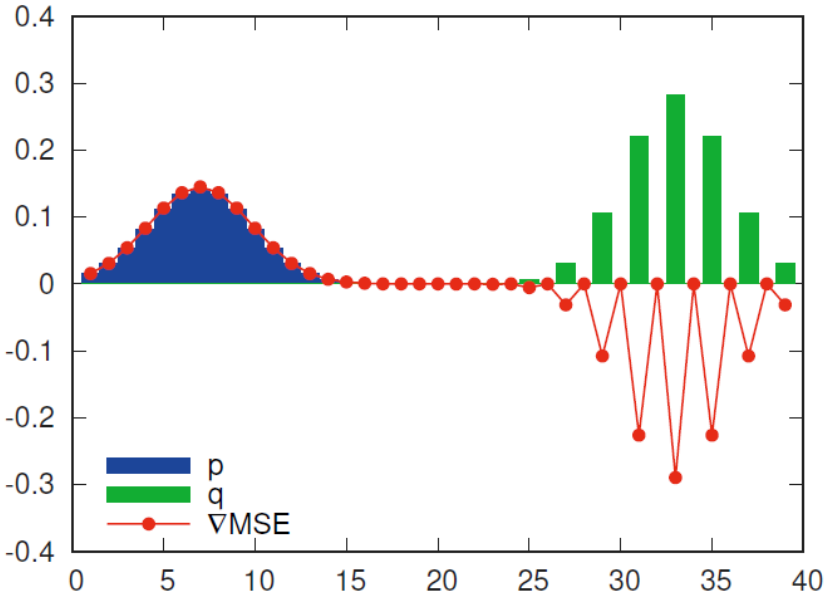
EMD: 50

- Closed form solution for EMD: $\sum_{i=1}^{N} |\varphi_i|, \text{ where } \varphi_i = \sum_{j=1}^{i} \left(\frac{a_j}{\|\mathbf{a}\|_1} - \frac{b_j}{\|\mathbf{b}\|_1} \right)$

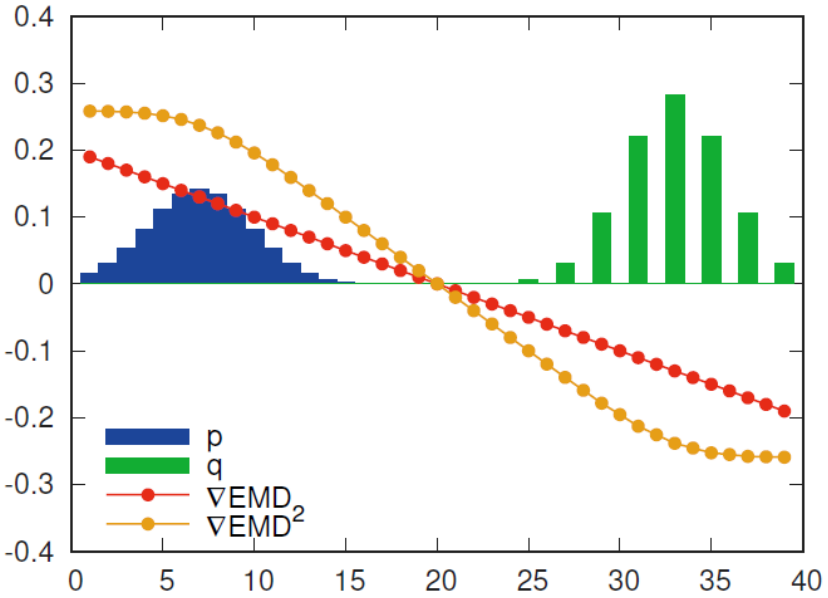
- Closed form for ∇EMD: $\sum_{i=1}^{N} \text{sgn}(\varphi_i) \sum_{j=1}^{i} (N\delta_{jk} - 1)$

"A Closed-form Gradient for the 1D Earth Mover's Distance for Spectral Deep Learning on Biological Data", Martinez, ICMLw 2016

How does the Gradient Look Like?

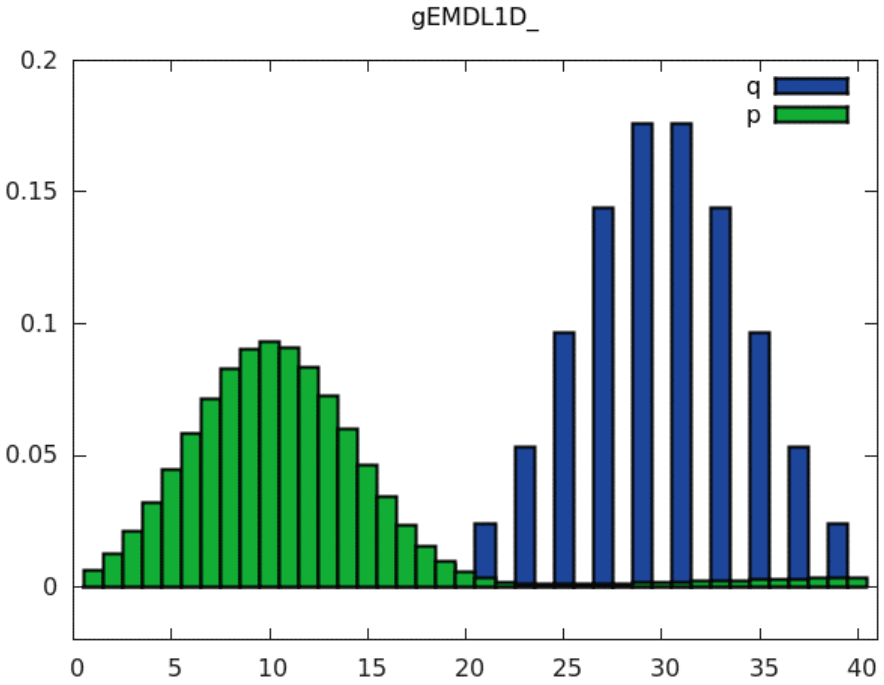


(a) Mean Squared Error

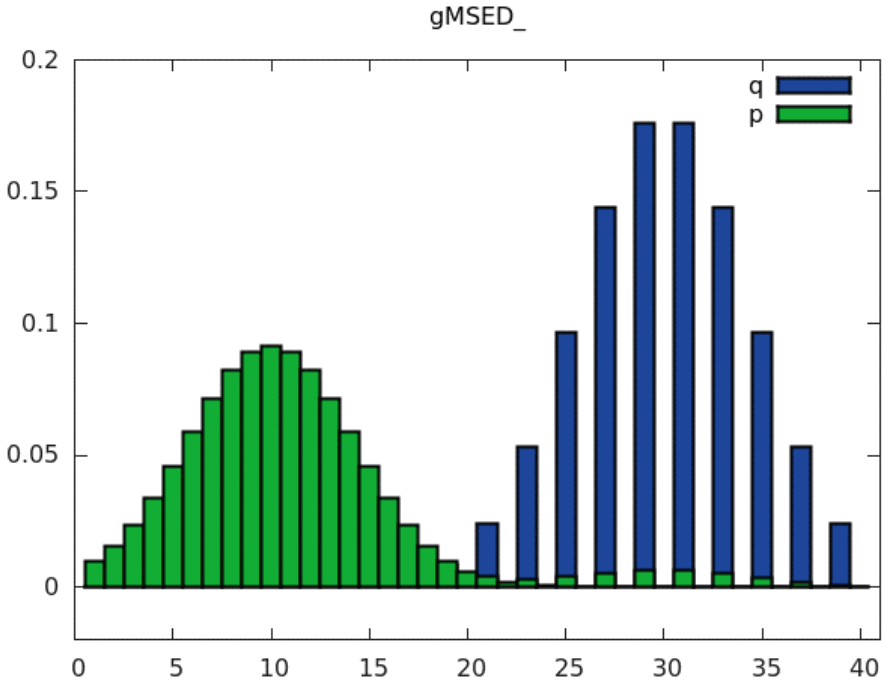


(b) Earth Mover's Distance

MSE vs Earth Mover's Distance: Smoothness



EMD



MSE

BACK TO THE GENERAL FORM

Sinkhorn Distance

Earth Mover's Distance: $$d_M(r,c) \stackrel{\text{def}}{=} \min_{P \in U(r,c)} \langle P, M \rangle.$$

Sinkhorn: $$d_M^\lambda(r,c) \stackrel{\text{def}}{=} \langle P^\lambda, M \rangle, \quad \text{where } P^\lambda = \operatorname*{argmin}_{P \in U(r,c)} \langle P, M \rangle - \frac{1}{\lambda} h(P).$$

Algorithm 1 Computation of $d_M^\lambda(r,c)$ using Sinkhorn-Knopp's fixed point iteration

```
Input M, λ, r, c.
I=(r>0); r=r(I); M=M(I,:); K=exp(-λ*M)
Set x=ones(length(r),size(c,2))/length(r);
while x changes do
  x=diag(1./r)*K*(c.*(1./(K'*(1./x))))
end while
u=1./x; v=c.*(1./(K'*u))
d_M^λ(r,c)=sum(u.*((K.*M)*v))
```

IT IS FAST!

Can we use it in DL? No! We need a gradient, not a distance.

"Sinkhorn distances: Lightspeed computation of optimal transport", Cuturi, NIPS 2013

Learning with a Wasserstein (sic.) Loss

Algorithm 1 Gradient of the Wasserstein loss

Given $h(x)$, y, λ, $\mathbf{K}$. (γ_a, γ_b if $h(x)$, y unnormalized.)
$u \leftarrow \mathbf{1}$
while u has not converged **do**

$$u \leftarrow \begin{cases} h(x) \oslash \left(\mathbf{K}\left(y \oslash \mathbf{K}^\top u\right)\right) & \text{if } h(x), y \text{ normalized} \\ \end{cases}$$

end while

$$\partial W_p^p / \partial h(x) \leftarrow \begin{cases} \frac{\log u}{\lambda} - \frac{\log u^\top \mathbf{1}}{\lambda K}\mathbf{1} & \text{if } h(x), y \text{ normalized} \end{cases}$$

- Implementations ready for Caffe and Julia.

"Learning with a Wasserstein Loss", Frogner, NIPS 2015

Learning with a Wasserstein (sic.) Loss

Task: Predict tags.
Inner metric: word2vec similarity between tags.
Results: worse AUC, better semantic similarity.

(a) **Flickr user tags**: street, parade, dragon; **our proposals**: people, protest, parade; **baseline proposals**: music, car, band.

(b) **Flickr user tags**: water, boat, reflection, sunshine; **our proposals**: water, river, lake, summer; **baseline proposals**: river, water, club, nature.

Figure 7: Examples of images in the Flickr dataset. We show the groundtruth tags and as well as tags proposed by our algorithm and the baseline.

"Learning with a Wasserstein Loss", Frogner, NIPS 2015

Names before Frogner's paper:

- Wasserstein metric: $W_p(\mu,\nu) := \left(\inf_{\gamma\in\Gamma(\mu,\nu)} \int_{M\times M} d(x,y)^p \, \mathrm{d}\gamma(x,y)\right)^{1/p},$ (continuous, analytic)

- Earth Mover's Distance: (discrete, slow)

$$EMD(P,Q) = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} f_{i,j} d_{i,j}}{\sum_{i=1}^{m}\sum_{j=1}^{n} f_{i,j}}$$

$$d_M(r,c) \stackrel{\text{def}}{=} \min_{P\in U(r,c)} \langle P, M \rangle.$$

- Mallows Distance: no one cares.

- Sinkhorn Distance: (discrete, fast, approximate)

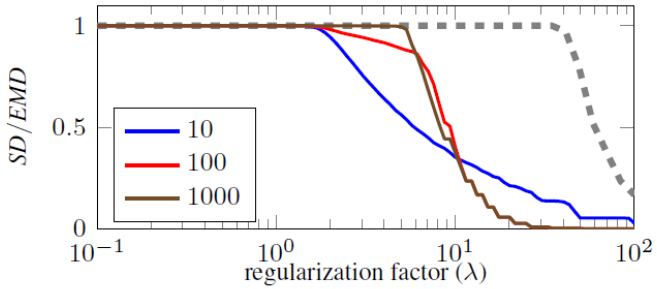
$$d_M^\lambda(r,c) \stackrel{\text{def}}{=} \langle P^\lambda, M \rangle, \quad \text{where } P^\lambda = \operatorname*{argmin}_{P\in U(r,c)} \langle P, M \rangle - \frac{1}{\lambda} h(P).$$

Names after Frogner's paper:

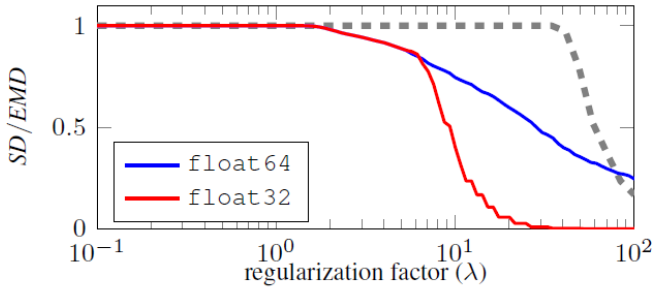
- Wasserstein metric: -> Wasserstein Distance (continuous, analytic)
- Earth Mover's Distance: -> Wasserstein Distance (discrete, slow)
- Mallows Distance: no one cares.
- Sinkhorn Distance: ->Wasserstein Distance (discrete, fast, approximate)

Problems of the Sinkhorn distance.

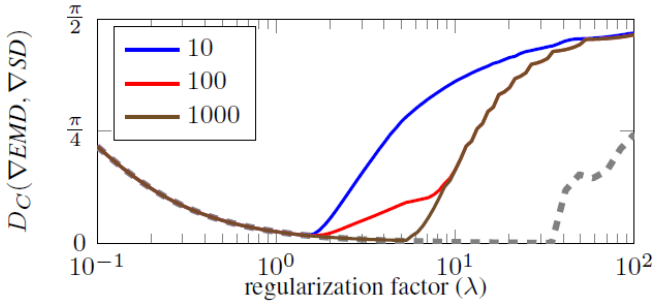
- It is still not that fast.
- EMD is a L1 distance (i.e., like Euclidean).
- Sinkhorn can be numerically unstable, particularly on GPUs:



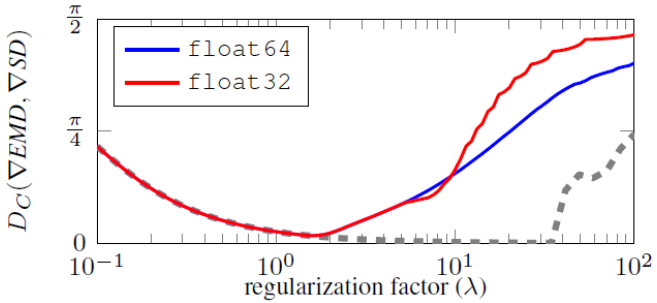
(a) Ratio of distances vs. Sinkhorn-Knopp iterations



(b) Ratio of distances vs. Floating point representation



(c) Angle between gradients vs. Sinkhorn-Knopp iterations



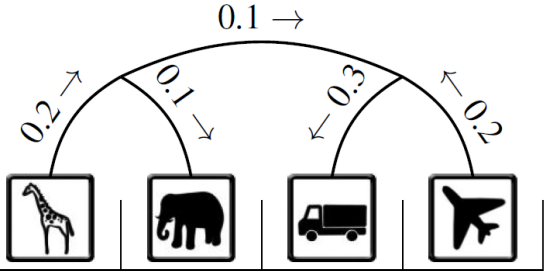
(d) Angle between gradients vs. Floating point representation

"Relaxed Earth Mover's Distances for Chain-and Tree-connected Spaces and their use as a Loss Function in Deep Learning", Martinez, arxiv, 2016

Possible solutions:

- Train with a L1 conscious algorithm. Lasso, lBFGS-OWL, etc…
  - Almost all Cuturi and Solomon papers.
- Use λ carefully to regularize Sinkhorn.
  - Fast Computation of Wasserstein Barycenters, Cuturi, 2014
- Explicit regularization.
  - Squared Earth Mover's Distance
    (only applicable to certain spaces)
- Other: Wasserstein GANs

Relaxed Earth Mover's Distance



f ()	0.2	0.4	0.2	0.2
y 	0.0	0.5	0.5	0.0
∇EMD	1.5	−0.5	−1.5	0.5
∇EMD^2	0.5	−0.1	−0.7	0.3

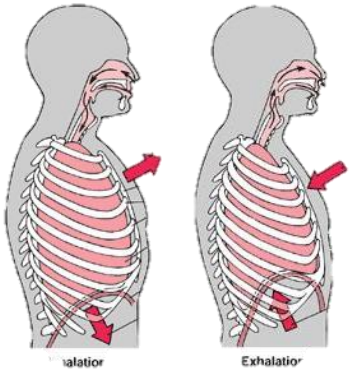
$$HEMD^{\rho}(\mathbf{p},\mathbf{q}) = \sum_{i \in G} M_{i,\mathbf{p}(i)} \cdot |\tilde{\varphi}_i|^{\rho},$$

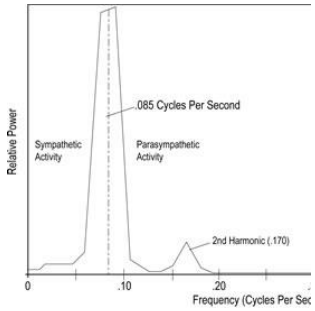
$$\tilde{\varphi}_i = \sum_{j \in \mathrm{I}(i)} (p_j - q_j) \ ,$$

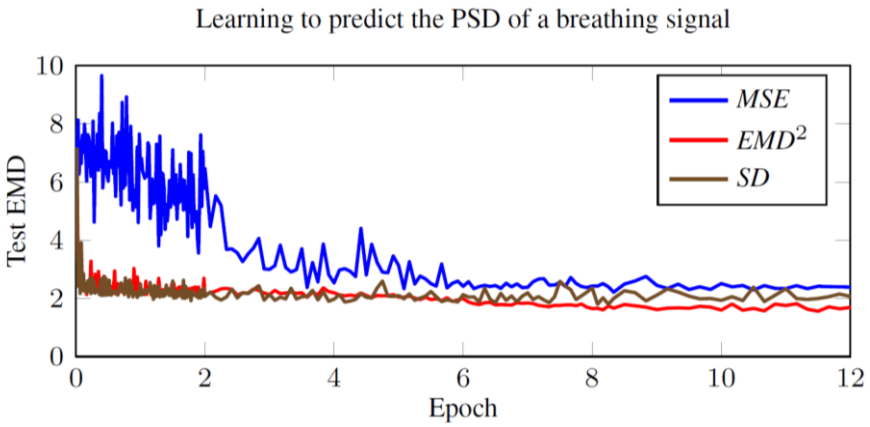
$$\nabla HEMD^{\rho} \simeq \rho \sum_{i \in G} \tilde{M}_i \cdot \tilde{\varphi}_i \cdot |\tilde{\varphi}_i|^{\rho-2} \sum_{j=1}^{i} (\delta_{jk} - 1/N)$$

"Relaxed Earth Mover's Distances for Chain-and Tree-connected Spaces and their use as a Loss Function in Deep Learning", Martinez, arxiv, 2016

Example: Breathing Rate

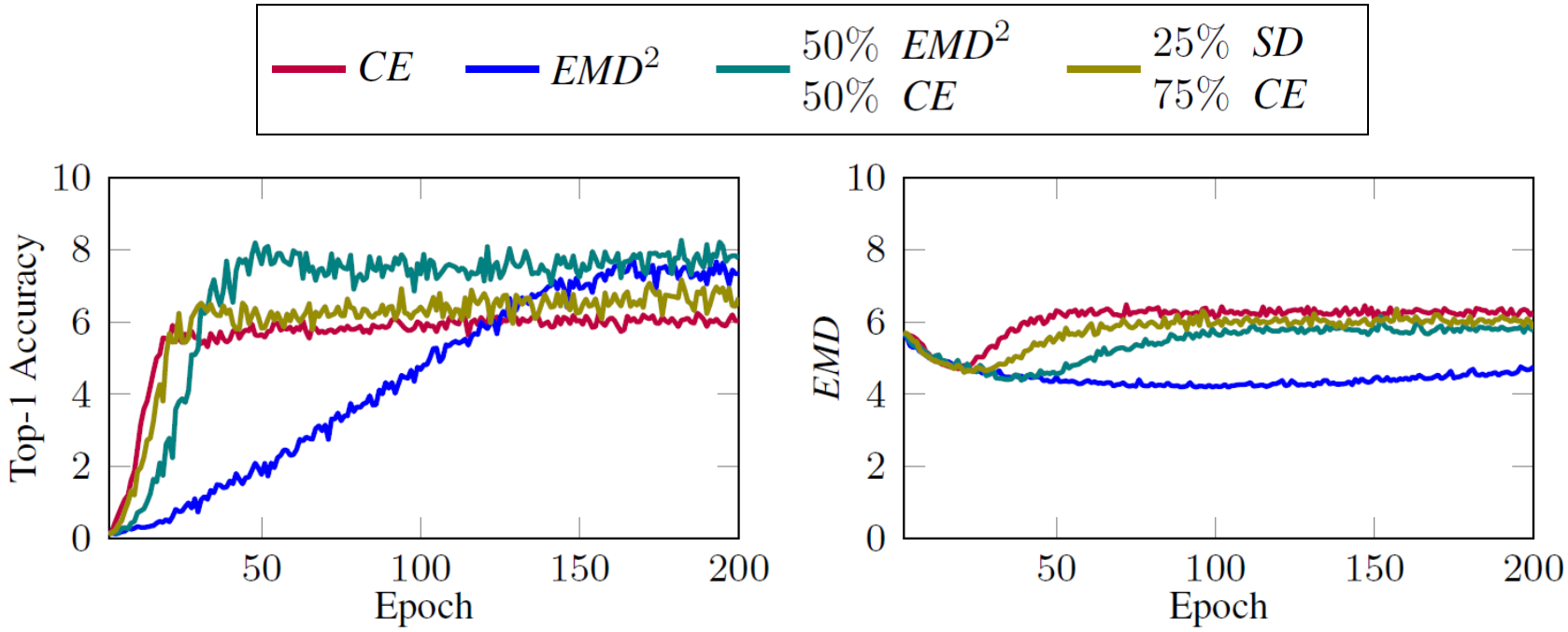






"Relaxed Earth Mover's Distances for Chain-and Tree-connected Spaces and their use as a Loss Function in Deep Learning", Martinez, arxiv, 2016

Example: ImageNet



(b) Top-1 Accuracy on ImageNet@50K images

(c) Holistic Error on ImageNet@50K images

“Relaxed Earth Mover's Distances for Chain-and Tree-connected Spaces and their use as a Loss Function in Deep Learning”, Martinez, arxiv, 2016

Conclusions:

- Earth Mover's Distance is a tricky Loss to use but:
  - It has become way more available recently.
  - It allows us to Embed a Metric from the Output Space.
  - It is Natural for Histogram Prediction / Probability Distributions, etc.

- Resources:
  - Math: http://marcocuturi.net/SI.html
  - Deep Learning: http://cbcl.mit.edu/wasserstein/
  - Fancy but broken uses:
    - Wasserstein GAN
    - Improved Training of Wasserstein GANs
    - Improving the Improved Training of Wasserstein GANs