

Reward-modulated inference

Buck Shlegeris Matthew Alger

COMP3740, 2014

Outline

- Supervised, unsupervised, and reinforcement learning
- Neural nets
- RMI
- Results with RMI

Types of machine learning

- supervised
- unsupervised
- reinforcement

MNIST



Supervised learning

Given some pairs $(x, f(x))$, approximate f .

- classification
- regression
- prediction

Unsupervised learning

Given data, find patterns.

- Goal is to increase likelihood of observed data.
- Related to compression
- This is useful as a preprocessing step.

Reinforcement learning

You see some stuff, and get some reward. What do you want to do?

- Way more general \rightarrow much harder \rightarrow make assumptions.
 - Stationary
 - MDP
- Value estimation? (Use supervised prediction?)
- Explore vs exploit?
- Preprocessing somehow?

What's a nice class of functions from \mathbb{R}^n to \mathbb{R}^m ?

$$f(\vec{x}) = W\vec{x} + \vec{b} \quad (1)$$

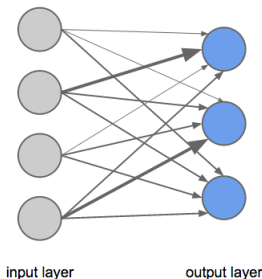
Let's use the name θ for our model, the combination of W and \vec{b} .

Logistic regression

$$P(Y = i | \theta, \vec{x}) = s(W\vec{x} + \vec{b})_i \quad (2)$$

(s rescales vectors in \mathbb{R}^n to have an L1 norm of 1.)

$$\text{prediction}(\theta, \vec{x}) = \underset{i}{\operatorname{argmax}}(\mathcal{P}(Y = i | \theta, \vec{x})) \quad (3)$$



$$\mathcal{L}(\theta, \vec{x}, y) = -\log(s(W\vec{x} + \vec{b})_y) \quad (4)$$

Overfitting?

$$\mathcal{L}(\theta, \vec{x}, y) = -\log(s(W\vec{x} + \vec{b})_y) - R(\theta) \quad (5)$$

What if instead of a single input vector \vec{x} and single label y , we had a whole list of inputs \mathcal{D} and a vector of labels \vec{y} ?

$$\mathcal{L}(\theta, \mathcal{D}, \vec{y}) = \sum_{i \in |\mathcal{D}|} \mathcal{L}(\theta, \mathcal{D}_i, \vec{y}_i) - R(\theta) \quad (6)$$

Logistic regression

$$\theta^*(\mathcal{D}, \vec{y}) = \underset{\theta}{\operatorname{argmin}} (\mathcal{L}(\theta, \mathcal{D}, \vec{y})) \quad (7)$$

$\theta \in (\mathbb{R}^{|x| \cdot |y|} \times \mathbb{R}^{|y|})$, so good luck finding that analytically

Logistic regression



Buck's Amazon.com

Today's Deals

Gift Cards

Sell

Help

Shop by
Department ▾

Search

Books ▾

mathematical optimization

Go

Hello, Buck
Your Account ▾

Your
Prime

Books

Advanced Search

New Releases

Best Sellers

The New York Times® Best Sellers

Children's Books

Textbooks

Textbook Ren

1-12 of 11,759 results for **Books** : "mathematical optimization"

Sort by [Re

Show results for

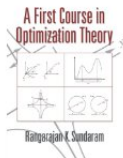
(Any Category

Books

- Mathematical Analysis (1,054)
- Discrete Mathematics (507)
- Operations Research (1,205)
- Game Theory (411)
- Theory of Economics (347)
- Econometrics (249)
- Mechanical Engineering (1,330)
- Mathematics (7,060)
- Economic History (31)
- Engineering (3,886)
- Science & Math (8,463)
- Engineering & Transportation (3,908)
- Business & Money (2,938)

+ See more

Book Format: Hardcover | Kindle Edition | Paperback



A First Course in Optimization Theory

Apr 3, 2014

by Rangarajan K. Sundaram

Paperback

\$21.00 to rent ✓Prime

\$42.75 to buy ✓Prime

Get it by **Tuesday, Oct 28**

More Buying Choices

\$16.79 used & new (61 offers)

Kindle Edition

\$28.91

Auto-delivered wirelessly

Other Formats: Hardcover

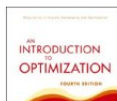


#1 Best Seller

Trade-in eligi

Excerpt

... lays the m
study of opti



An Introduction to Optimization

Jan 14, 2013

by Edwin K. P. Chong and Stanislaw H. Zak

Hardcover

\$82.66 ~~\$449.00~~ ✓Prime

Only 4 left in stock - order soon.



Trade-in eligi

Buck Shlegeris, Matthew Alger

Reward-modulated inference

Logistic regression

Gradient descent

while last update was bigger than ϵ **do**

$$W_{\text{new}} \leftarrow W - \alpha \frac{\partial \mathcal{L}(\theta, \mathcal{D}, \vec{y})}{\partial W}$$

end while

Logistic regression

Stochastic gradient descent

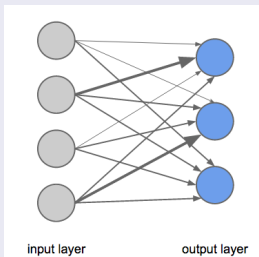
```
for input  $\vec{x}$  and label  $y \in (\mathcal{D}, \vec{y})$  do  
     $w \leftarrow W - \alpha \frac{\partial \mathcal{L}(\theta, \vec{x}, y)}{\partial W}$   
end for
```

Logistic regression

Only linearly separable things can be separated!

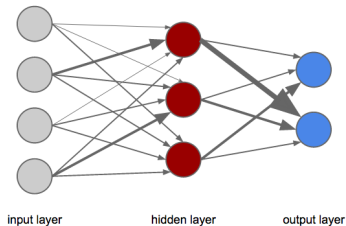
Multilayer perceptron

Logistic regression



$$s(W\vec{x} + \vec{b})$$

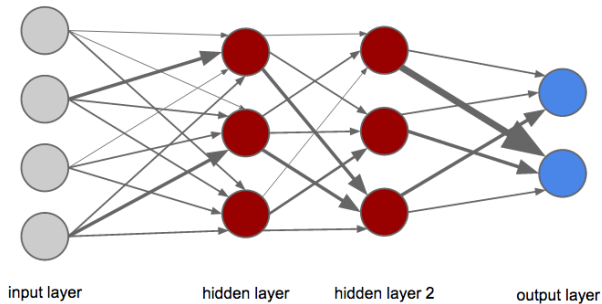
Multilayer perceptron



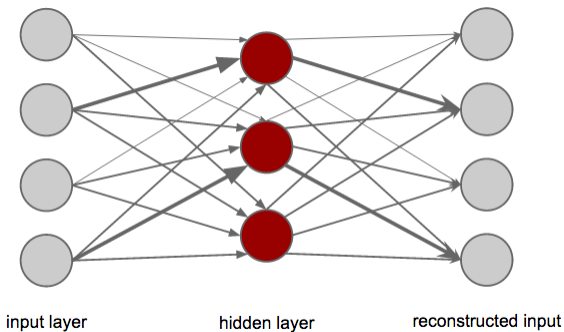
$$s(Ws(W_2\vec{x} + \vec{b}_2) + \vec{b}) \\ = s(W\vec{h} + \vec{b})$$

Multilayer perceptron

Why stop at 1 layer?



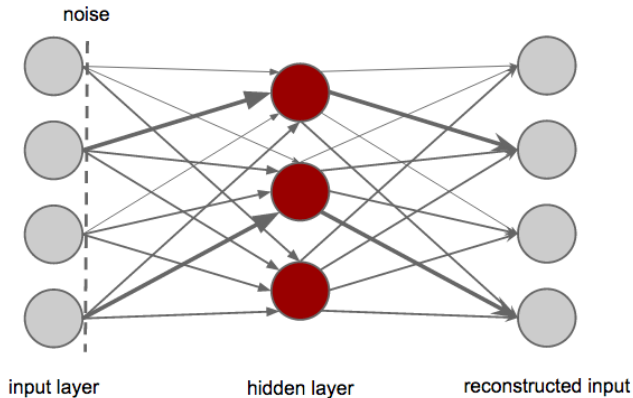
Autoencoders



$$\vec{x}' = s(W \cdot s(W_2 \vec{x} + \vec{b}_2) + \vec{b}) \quad (8)$$

$$\mathcal{L}(\theta, \vec{x}) = ||\vec{x} - \vec{x}'|| \quad (9)$$

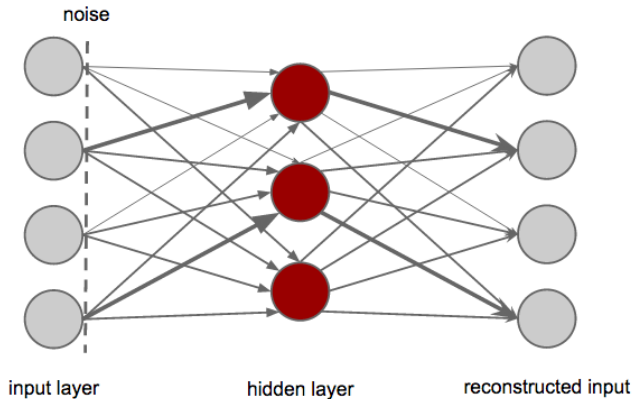
Denoising autoencoders (dAs)



$$s(W \cdot s(W_2 n(\vec{x}) + \vec{b}_2) + \vec{b}) \quad (10)$$

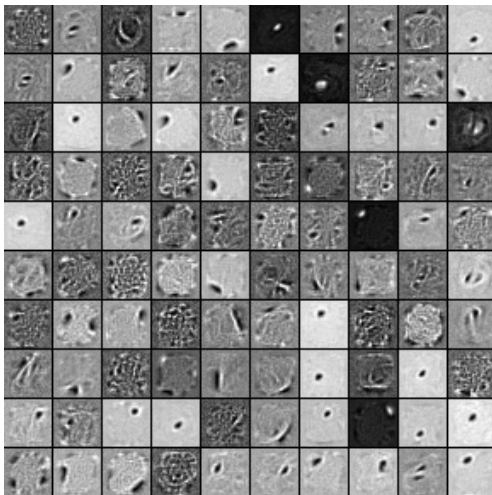
(where n is a stochastic noise function)

Denoising autoencoders (dAs)

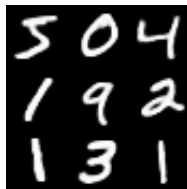
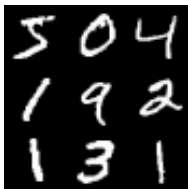


$$s(W \cdot s(W_2 \vec{x} + \vec{b}_2) + \vec{b}) \quad (11)$$

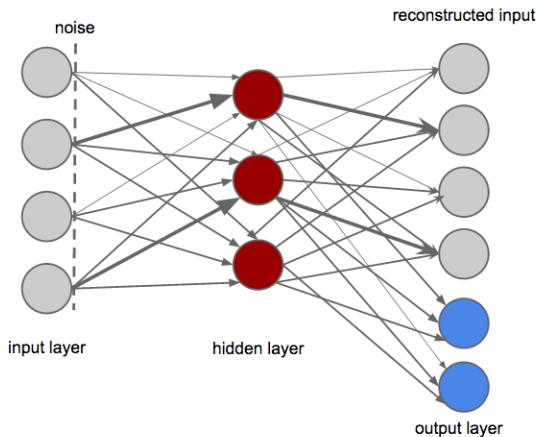
Denoising autoencoders (dAs)



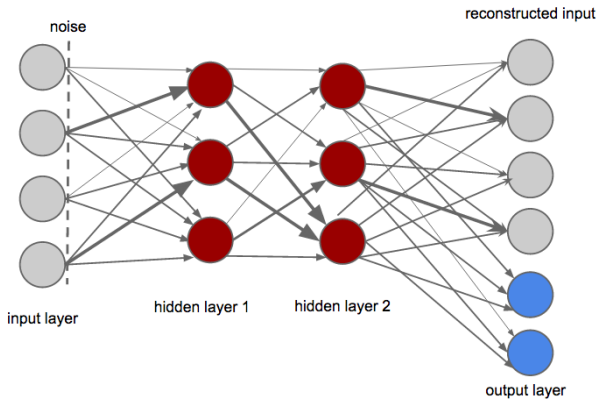
Denoising autoencoders (dAs)



Denoising autoencoders (dAs)



Denoising autoencoders (dAs)



Denoising autoencoders (dAs)

$$\mathcal{L}(\theta, \vec{x}, y) = -\log(s(W\vec{x} + \vec{b})_y) \quad (12)$$

$$\mathcal{L}(\theta, \vec{x}) = \|\vec{x} - \vec{x}'\| \quad (13)$$

How do we mix between these?

Reward modulation

Add a time-varying modulation function $\lambda(t)$:

$$\mathcal{L}(\theta, \vec{x}, y) = \lambda(t) \cdot \text{supervised cost} + (1 - \lambda(t)) \cdot \text{unsupervised cost} \quad (14)$$

$$\mathcal{L}(\theta, \vec{x}, y) = \lambda(t) \left(-\log(s(W\vec{x} + \vec{b})_y) \right) + (1 - \lambda(t)) \left(\|\vec{x} - \vec{x}'\| \right) \quad (15)$$

Reward modulation

Motivations:

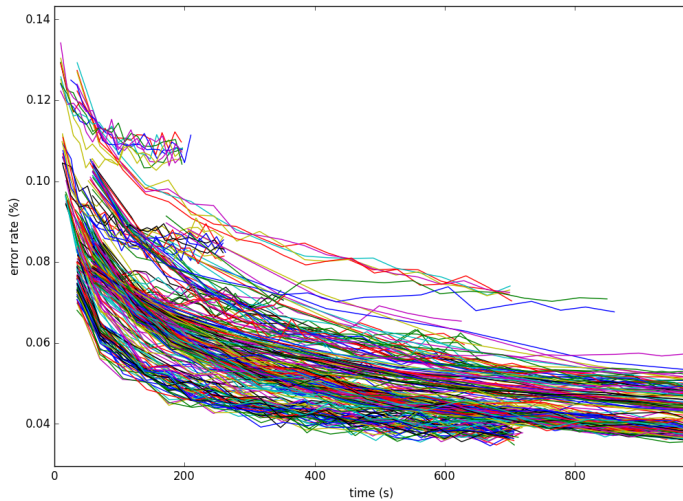
- Information theory
- Fine tuning
- Extreme learning

Reward modulation

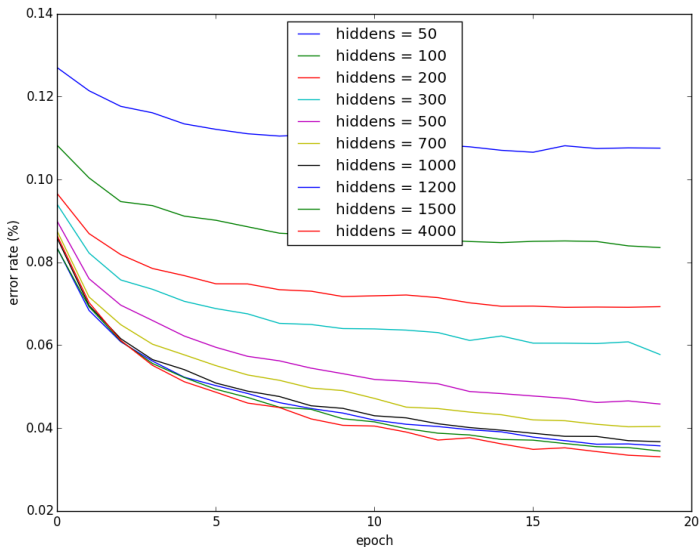
Questions:

- Does it improve performance on problems?
- How should we vary reward modulation?

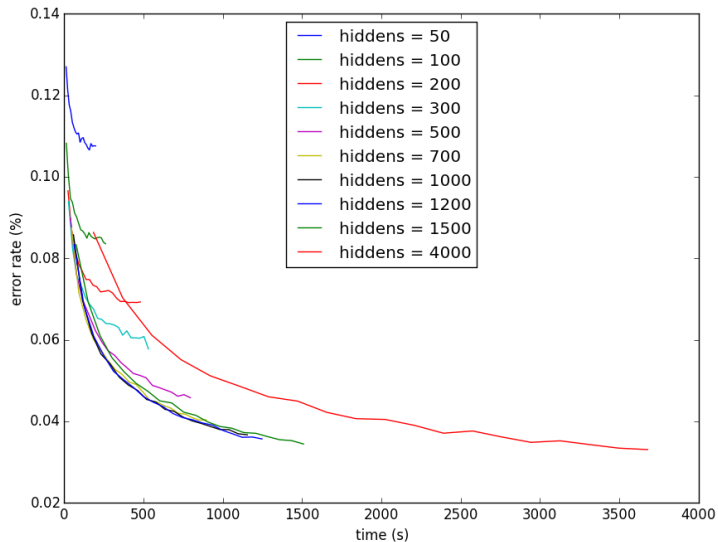
Results



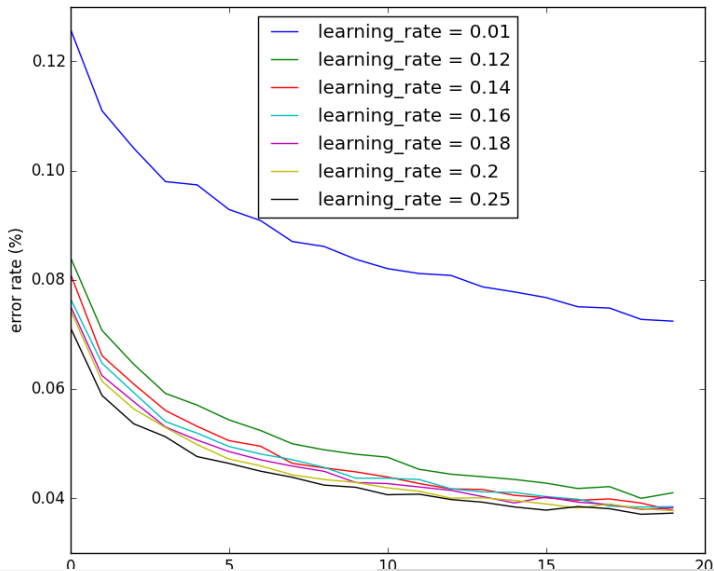
Hyperparameters



Hyperparameters



Hyperparameters



Classification problem

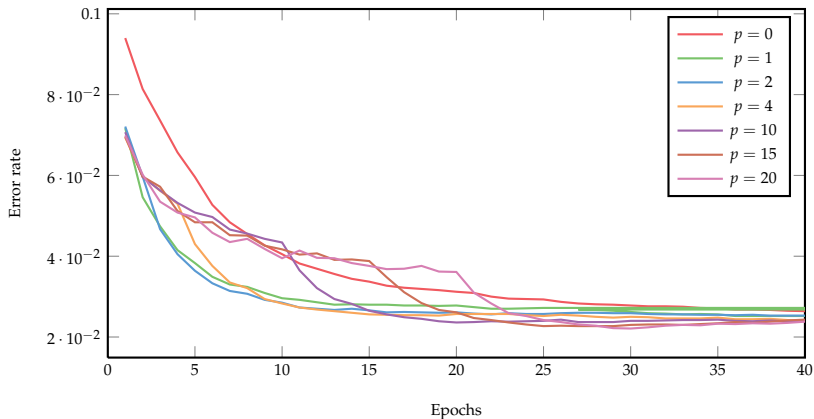


Figure: Step reward modulation with $\lambda = 0$ if $t < p$, and $\lambda = 0$ otherwise.

Classification problem

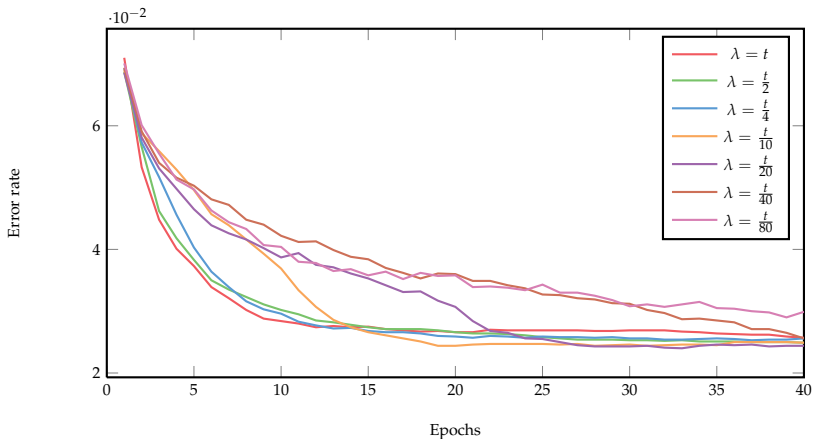


Figure: Linear reward modulation with different changes in λ .

Classification problem

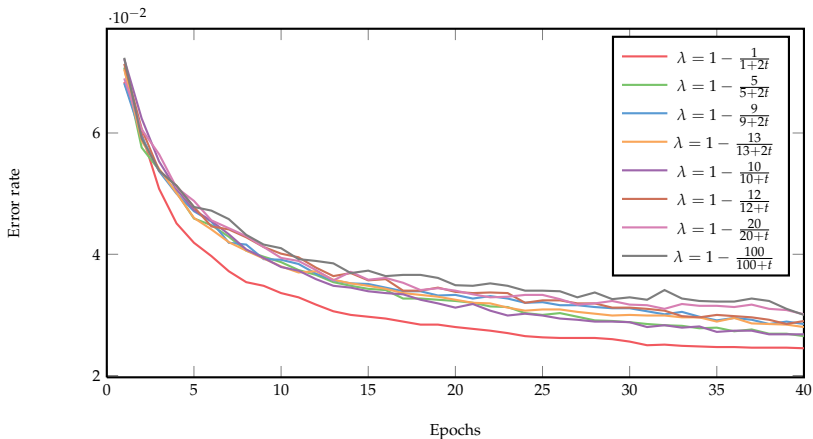
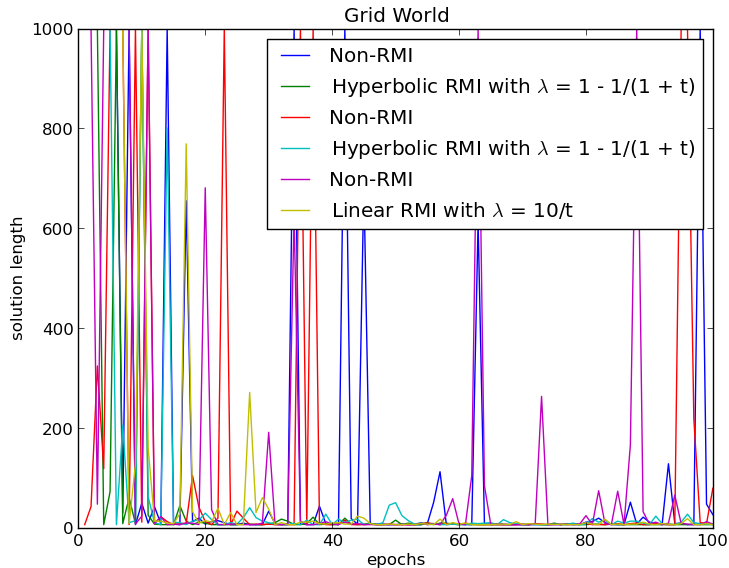


Figure: Hyperbolic reward modulation with different changes in λ .

no results yet.



Conclusions

- RMI works on classification (maybe because it's like fine-tuning)
- Haven't got contextual bandit results yet.
- Works nicely on grid world for some reason, more MDP data to come.