Hope you had a FANTASTIC spring break!
Hope you had a FANTASTIC spring break!

Thanksgiving
CS 188: Artificial Intelligence

Neural Nets (ctd) and IRL

Instructor: Anca Dragan --- University of California, Berkeley

[These slides were created by Dan Klein and Pieter Abbeel for CS188 Intro to AI at UC Berkeley. All CS188 materials are available at http://ai.berkeley.edu.]
Reminder: Linear Classifiers

- Inputs are feature values
- Each feature has a weight
- Sum is the activation

\[
\text{activation}_w(x) = \sum_i w_i \cdot f_i(x) = w \cdot f(x)
\]

- If the activation is:
  - Positive, output +1
  - Negative, output -1
Multiclass Logistic Regression

- Multi-class linear classification
  - A weight vector for each class: $w_y$
  - Score (activation) of a class $y$: $w_y \cdot f(x)$
  - Prediction w/highest score wins: $y = \text{arg max}_y w_y \cdot f(x)$

- How to make the scores into probabilities?

$$z_1, z_2, z_3 \rightarrow \frac{e^{z_1}}{e^{z_1} + e^{z_2} + e^{z_3}}, \frac{e^{z_2}}{e^{z_1} + e^{z_2} + e^{z_3}}, \frac{e^{z_3}}{e^{z_1} + e^{z_2} + e^{z_3}}$$

original activations

softmax activations
Best w?

- Maximum likelihood estimation:

\[
\max_w \quad ll(w) = \max_w \quad \sum_i \log P(y^{(i)} | x^{(i)} ; w)
\]

with:

\[
P(y^{(i)} | x^{(i)} ; w) = \frac{e^{w_y(i) \cdot f(x^{(i)})}}{\sum_y e^{w_y \cdot f(x^{(i)})}}
\]

= Multi-Class Logistic Regression
Gradient in n dimensions

\[ \nabla g = \begin{bmatrix} \frac{\partial g}{\partial w_1} \\ \frac{\partial g}{\partial w_2} \\ \vdots \\ \frac{\partial g}{\partial w_n} \end{bmatrix} \]
Optimization Procedure: Gradient Ascent

- init $w$
- for iter = 1, 2, ...

$$w \leftarrow w + \alpha \cdot \nabla g(w)$$

- $\alpha$: learning rate --- tweaking parameter that needs to be chosen carefully
- How? Try multiple choices
  - Crude rule of thumb: update changes $w$ about 0.1 – 1%
Neural Networks
Multi-class Logistic Regression

- special case of neural network

\[
P(y_1|x; w) = \frac{e^{z_1}}{e^{z_1} + e^{z_2} + e^{z_3}}
\]

\[
P(y_2|x; w) = \frac{e^{z_2}}{e^{z_1} + e^{z_2} + e^{z_3}}
\]

\[
P(y_3|x; w) = \frac{e^{z_3}}{e^{z_1} + e^{z_2} + e^{z_3}}
\]
Deep Neural Network = Also learn the features!

\[
z_i^{(k)} = g \left( \sum_j W_{i,j}^{(k-1,k)} z_j^{(k-1)} \right)
\]

\[g = \text{nonlinear activation function}\]
Training the deep neural network is just like logistic regression: 

$$\max_w \; ll(w) = \max_w \sum_i \log P(y^{(i)}|x^{(i)}; w)$$

just w tends to be a much, much larger vector 😊

→ just run gradient ascent
+ stop when log likelihood of hold-out data starts to decrease
How well does it work?
Computer Vision
Object Detection
Manual Feature Design
Features and Generalization

[HoG: Dalal and Triggs, 2005]
Features and Generalization

Image

HoG
Performance

ImageNet Error Rate 2010-2014

Error Rate

79%
60%
40%
20%
7%

2010 2011 2012 2013 2014

Traditional CV

graph credit Matt Zeiler, Clarifai
Performance

ImageNet Error Rate 2010-2014

- Traditional CV
- Deep Learning

AlexNet

Graph credit Matt Zeiler, Clarifai
Performance

ImageNet Error Rate 2010-2014

- Traditional CV
- Deep Learning

AlexNet

Graph credit: Matt Zeiler, Clarifai
MS COCO Image Captioning Challenge

"man in black shirt is playing guitar."
"construction worker in orange safety vest is working on road."
"two young girls are playing with lego toy."
"boy is doing backflip on wakeboard."
"girl in pink dress is jumping in air."
"black and white dog jumps over bar."
"young girl in pink shirt is swinging on swing."
"man in blue wetsuit is surfing on wave."

Karpathy & Fei-Fei, 2015; Donahue et al., 2015; Xu et al, 2015; many more
Visual QA Challenge
Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, Devi Parikh

What vegetable is on the plate?  
Neural Net: **broccoli**  
Ground Truth: broccoli

What color are the shoes on the person's feet?  
Neural Net: **brown**  
Ground Truth: brown

How many school busses are there?  
Neural Net: **2**  
Ground Truth: 2

What sport is this?  
Neural Net: **baseball**  
Ground Truth: baseball

What is on top of the refrigerator?  
Neural Net: **magnets**  
Ground Truth: cereal

What uniform is she wearing?  
Neural Net: **shorts**  
Ground Truth: girl scout

What is the table number?  
Neural Net: **4**  
Ground Truth: 40

What are people sitting under in the back?  
Neural Net: **bench**  
Ground Truth: tent
Speech Recognition

TIMIT Speech Recognition

Error Rate

- Traditional
- Deep Learning

graph credit Matt Zeiler, Clarifai
Machine Translation

Google Neural Machine Translation (in production)
What’s still missing? – correlation \neq causation

Figure 11: Raw data and explanation of a bad model’s prediction in the “Husky vs Wolf” task.

<table>
<thead>
<tr>
<th></th>
<th>Before</th>
<th>After</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trusted the bad model</td>
<td>10 out of 27</td>
<td>3 out of 27</td>
</tr>
<tr>
<td>Snow as a potential feature</td>
<td>12 out of 27</td>
<td>25 out of 27</td>
</tr>
</tbody>
</table>

Table 2: “Husky vs Wolf” experiment results.
What’s still missing? – covariate shift

[Carroll et al.]
What’s still missing? – covariate shift

[Carroll et al.]
What’s still missing – knowing what loss to optimize
Neural Nets (ctd) and IRL

Instructor: Anca Dragan --- University of California, Berkeley

[These slides were created by Dan Klein and Pieter Abbeel for CS188 Intro to AI at UC Berkeley. All CS188 materials are available at http://ai.berkeley.edu.]
Reminder: Optimal Policies

R(s) = -0.01
R(s) = -0.03
R(s) = -0.4
R(s) = -2.0
Utility?

Clear utility function

Not so clear utility function
$R \rightarrow \pi^*$
Inverse Planning/RL

\[ \pi^* \rightarrow R \]
$\xi \rightarrow R$
Inverse Planning/RL
Inverse Planning/RL
IRL is relevant to all 3 types of people:

- person in its environment
- its end-user
- its designer
Inverse Planning/RL

given: \( \xi_D \)

find: \( R(s, a) \)

s.t. \( R(\xi_D) \geq R(\xi) \forall \xi \)
Inverse Planning/RL

given: \( \xi_D \)

find: \( R(s, a) = \theta^T \phi(s, a) \)

s.t. \( R(\xi_D) \geq R(\xi) \forall \xi \)
Inverse Planning/RL

given: $\check{\xi}_D$

find: $R(s, a) = \theta^T \phi(s, a)$

s.t. $R(\check{\xi}_D) \geq \max_{\xi} R(\xi)$
Problem

given: \( \xi_D \) 

zero/constant reward is a solution

find: 

\[
R(s, a) = \theta^T \phi(s, a)
\]

s.t. 

\[
R(\xi_D) \geq \max_\xi R(\xi)
\]
Revised formulation

given: \( \xi_D \)

find: \( R(s, a) = \theta^T \phi(s, a) \)

s.t. \( R(\xi_D) \geq \max_\xi [R(\xi) + l(\xi, \xi_D)] \)

small close to the demonstration
\[
\max_{\theta} \left[ R(\xi_D) - \max_{\xi} \left[ R(\xi) + l(\xi, \xi_D) \right] \right]
\]
$\max_{\theta} \left[ \theta^T \phi(\xi_D) - \max_{\xi} \left[ \theta^T \phi(\xi) + l(\xi, \xi_D) \right] \right]$
Optimization

\[ \xi_\theta^* = \arg \max \max_\theta [\theta^T \phi(\xi_D) - \max_\xi [\theta^T \phi(\xi) + l(\xi, \xi_D)]] \]
Optimization

$$\max_{\theta} \left[ \theta^T \phi(\xi_D) - \max_{\xi} \left[ \theta^T \phi(\xi) + l(\xi, \xi_D) \right] \right]$$

subgradient: \[ \nabla_{\theta} = \phi(\xi_D) - \phi(\xi_{\theta}^*) \]
Optimization

\[
\max_{\theta} [\theta^T \phi(\xi_D) - \max_{\xi} [\theta^T \phi(\xi) + l(\xi, \xi_D)]]
\]

subgradient:

\[
\nabla_{\theta} = \phi(\xi_D) - \phi(\xi^{*}_{\theta})
\]

\[
\theta_{i+1} = \theta_i + \alpha (\phi(\xi_D) - \phi(\xi^{*}_{\theta_i}))
\]
Interpretation

\[ \phi(\xi^*_i ) \quad \text{goes on rocks: } [1,0] \]

\[ \phi(\xi_D) \quad \text{goes on grass: } [0,1] \]

\[ \theta_{i+1} = \theta_i + \alpha(\phi(\xi_D) - \phi(\xi^*_i)) \]
Interpretation

\[ \phi(\xi_{\theta_i}) \] goes on rocks: [1,0]

\[ \phi(\xi_D) \] goes on grass: [0,1]

\[ \theta_{i+1} = \theta_i + \alpha([-1,1]) \]

\[ \theta_{i+1} = \theta_i + \alpha(\phi(\xi_D) - \phi(\xi_{\theta_i})) \]
Interpretation

\( \phi(\xi^*_i) \)  goes on rocks: [1,0]

\( \phi(\xi_D) \)  goes on grass: [0,1]

rocks weight goes down  grass weight goes up

\[
\theta_{i+1} = \theta_i + \alpha([\!-1,1])
\]

\[
\theta_{i+1} = \theta_i + \alpha(\phi(\xi_D) - \phi(\xi_{\theta_i}^*))
\]
Interpretation

\[ \phi(\tilde{\xi}_i^*) \text{ goes on rocks: [1,0]} \]

\[ \phi(\tilde{\xi}_D) \text{ goes on grass: [0,1]} \]

rocks weight goes down  grass weight goes up

The new reward likes grass more and rocks less.
Inverse Planning/RL
Inverse Planning/RL
Is the demonstrator really optimal?

\[ R(\xi_D) \geq R(\xi) \forall \xi \]
The Bayesian view

\[ P(\xi_D | \theta) \]

evidence hidden
The Bayesian view

\[ P(\xi_D | \theta) \propto e^{\beta \theta^T \phi(\xi_D)} \]
The Bayesian view

\[ P(\xi_D | \theta) = \frac{e^{\beta \theta^T \phi(\xi_D)}}{\sum_\xi e^{\beta \theta^T \phi(\xi)}} \]
The Bayesian view

\[ P(\xi_D | \theta) = \frac{e^{\beta \theta^T \phi(\xi_D)}}{\sum_{\xi} e^{\beta \theta^T \phi(\xi)}} \]

\[ b'(\theta) \propto b(\theta) P(\xi_D | \theta) \]
The Bayesian view

\[ P(\xi_D | \theta) = \frac{e^{\beta \theta^T \phi(\xi_D)}}{\sum_{\xi} e^{\beta \theta^T \phi(\xi)}} \]

\[ \max_{\theta} P(\xi_D | \theta) \]
The Bayesian view

$$\max_{\theta} \log \frac{e^{\beta \theta^T \phi(\xi_D)}}{\sum_{\xi} e^{\beta \theta^T \phi(\xi)}}$$
The Bayesian view

\[
\max_{\theta} \beta \theta^T \phi(\xi_D) - \log \sum_{\xi} e^{\beta \theta^T \phi(\xi)}
\]
The Bayesian view

$$\max_{\theta} \beta \theta^T \phi(\xi_D) - \log \sum_{\xi} e^{\beta \theta^T \phi(\xi)}$$

$$\nabla_\theta = \beta \phi(\xi_D) - \frac{1}{\sum_{\xi} e^{\beta \theta^T \phi(\xi)}}$$
The Bayesian view

$$\max_{\theta} \beta \theta^T \phi(\xi_D) - \log \sum_{\xi} e^{\beta \theta^T \phi(\xi)}$$

$$\nabla_{\theta} = \beta \phi(\xi_D) - \frac{1}{\sum_{\xi} e^{\beta \theta^T \phi(\xi)}} \nabla(\sum_{\xi} e^{\beta \theta^T \phi(\xi)})$$
The Bayesian view

\[
\max_{\theta} \beta \theta^T \phi(\xi_D) - \log \sum_{\xi} e^{\beta \theta^T \phi(\xi)}
\]

\[
\nabla_{\theta} = \beta \phi(\xi_D) - \frac{1}{\sum_{\xi} e^{\beta \theta^T \phi(\xi)}} \sum_{\xi} e^{\beta \theta^T \phi(\xi)} \beta \phi(\xi)
\]
The Bayesian view

$$\max_{\theta} \beta \theta^T \phi(\xi_D) - \log \sum_{\xi} e^{\beta \theta^T \phi(\xi)}$$

$$\nabla_{\theta} = \beta \phi(\xi_D) - \sum_{\xi} \frac{e^{\beta \theta^T \phi(\xi)}}{\sum_{\xi_i} e^{\beta \theta^T \phi(\xi_i)}} \beta \phi(\xi)$$
The Bayesian view

\[
\max_{\theta} \beta \theta^T \phi(\xi_D) - \log \sum_{\xi} e^{\beta \theta^T \phi(\xi)}
\]

\[
\nabla_\theta = \beta \phi(\xi_D) - \sum_{\xi} P(\xi|\theta) \beta \phi(\xi)
\]
The Bayesian view

$$\max_{\theta} \beta \theta^T \phi(\xi_D) - \log \sum_{\xi} e^{\beta \theta^T \phi(\xi)}$$

$$\nabla_{\theta} = \beta \left( \phi(\xi_D) - \mathbb{E}_{\xi \sim \theta} \phi(\xi) \right)$$
The Bayesian view

$$\max_{\theta} \beta \theta^T \phi(\xi_D) - \log \sum_{\xi} e^{\beta \theta^T \phi(\xi)}$$

$$\nabla_{\theta} = \beta (\phi(\xi_D) - \mathbb{E}_{\xi \sim \theta} \phi(\xi))$$

expected feature values produced by the current reward
The Bayesian view

\[ P(\xi_D | \theta) = \frac{e^{\beta \theta^T \phi(\xi_D)}}{\sum_{\xi} e^{\beta \theta^T \phi(\xi)}} \]

\[ b'(\theta) \propto b(\theta) P(\xi_D | \theta) \]
The Bayesian view (actions)

\[ P(\alpha_D | s, \theta) = \frac{e^{\beta Q(s, \alpha_D; \theta)}}{\sum_{\alpha} e^{\beta Q(s, \alpha; \theta)}} \]

\[ b'(\theta) \propto b(\theta) P(\alpha_D | \theta) \]
[Ratliff et al. *Maximum Margin Planning*]
[Levine et al. Continuous Inverse Optimal Control with Locally Linear Examples]
Continuous Inverse Optimal Control with Locally Optimal Examples

driving policies learned from human examples