Recall: Binary Classification

- Logistic regression: \( P(y = 1|x) = \frac{\exp(\sum_{i=1}^{n} w_i x_i)}{1 + \exp(\sum_{i=1}^{n} w_i x_i)} \)
  - Decision rule: \( P(y = 1|x) \geq 0.5 \iff w^T x \geq 0 \)
  - Gradient (unregularized): \( x(y - P(y = 1|x)) \)
- SVM: quadratic program to minimize weight vector norm w/slack
  - Decision rule: \( w^T x \geq 0 \)
  - (Sub)gradient (unregularized): 0 if correct with margin of 1, else \( x(2y - 1) \)

Loss Functions

- Hinge (SVM)
- 0-1 (ideal)
- Perceptron
- Logistic

\[ w^T x \]
This Lecture

- Multiclass fundamentals
- Feature extraction
- Multiclass logistic regression
- Multiclass SVM
- Optimization

Multiclass Fundamentals

Text Classification

- A Cancer Conundrum: Too Many Drug Trials, Too Few Patients
- Yankees and Mets Are on Opposite Tracks

- Health
- Sports
- ~20 classes

Image Classification

- Dog
- Car

- Thousands of classes (ImageNet)
Although he originally won the event, the United States Anti-Doping Agency announced in August 2012 that they had disqualified Armstrong from his seven consecutive Tour de France wins from 1999–2005.

Lance Edward Armstrong is an American former professional road cyclist

Armstrong County is a county in Pennsylvania...

4,500,000 classes (all articles in Wikipedia)

One day, James thought he would go into town and see what kind of trouble he could get into. He went to the grocery store and pulled all the pudding off the shelves and ate two jars. Then he walked to the fast food restaurant and ordered 15 bags of fries. He didn’t pay, and instead headed home.

3) Where did James go after he went to the grocery store?
A) his deck
B) his freezer
C) a fast food restaurant
D) his room

Multiple choice questions, 4 classes (but classes change per example)

Can we just use binary classifiers here?
Multiclass Classification

- One-vs-all: train $k$ classifiers, one to distinguish each class from all the rest
- How do we reconcile multiple positive predictions? Highest score?

- Not all classes may even be separable using this approach
- Can separate 1 from 2+3 and 2 from 1+3 but not 3 from the others (with these features)

- All-vs-all: train $(n-1)/2$ classifiers to differentiate each pair of classes
- Again, how to reconcile?

- Binary classification: one weight vector defines both classes
- Multiclass classification: different weights and/or features per class
Multiclass Classification

- Formally: instead of two labels, we have an output space $\mathcal{Y}$ containing a number of possible classes
- Same machinery that we’ll use later for exponentially large output spaces, including sequences and trees
- Decision rule: $\arg\max_{y \in \mathcal{Y}} w^T f(x, y)$
- Multiple feature vectors, one weight vector
- Can also have one weight vector per class: $\arg\max_{y \in \mathcal{Y}} w_y^T f(x)$
- Single-weight-vector is going to be better for reasons we’ll come back to

Feature Extraction

Block Feature Vectors

- Decision rule: $\arg\max_{y \in \mathcal{Y}} w^T f(x, y)$
- Base feature function:
  - $f(x) = I[\text{contains } \text{drug}], I[\text{contains } \text{patients}], I[\text{contains } \text{baseball}] = [1, 1, 0]$ for each label
  - $f(x, y = \text{Health}) = [1, 1, 0, 0, 0, 0, 0, 0]$ [contains drug & label = Health]
  - $f(x, y = \text{Sports}) = [0, 0, 0, 1, 1, 0, 0, 0]$ no drug
- Equivalent to having three weight vectors in this case

Making Decisions

- $f(x) = I[\text{contains drug}], I[\text{contains patients}], I[\text{contains baseball}]$
- $f(x, y = \text{Health}) = [1, 1, 0, 0, 0, 0, 0, 0]$ too many drug trials, too few patients
- $f(x, y = \text{Sports}) = [0, 0, 0, 1, 1, 0, 0, 0]$ too many drug trials, too few patients
- $w = [+2.1, +2.3, -5, -2.1, -3.8, +5.2, +1.1, -1.7, -1.3]$ word drug in Science article = +1.1
- $w^T f(x, y) = \text{Health: } +4.4$ too many drug trials, too few patients
- $w^T f(x, y) = \text{Sports: } -5.9$
- $w^T f(x, y) = \text{Science: } -1.9$
Another example: POS tagging

- Classify blocks as one of 36 POS tags
- Example x: sentence with a word (in this case, blocks) highlighted
- Extract features with respect to this word:
  \[ f(x, y_{\text{VBZ}}) = \begin{cases} 
  \text{I[curr_word=blocks \\ & tag = VBZ]}, \\
  \text{I[prev_word=router \\ & tag = VBZ]}, \\
  \text{I[next_word=the \\ & tag = VBZ]}, \\
  \text{I[curr_suffix=s \\ & tag = VBZ]}, \\
  \end{cases} \]
- Next two lectures: sequence labeling!

Multiclass Logistic Regression

- Training:
  - Multiclass logistic regression:
    \[
    P_w(y|x) = \frac{\exp(w^T f(x, y))}{\sum_{y' \in Y} \exp(w^T f(x, y'))}
    \]
  - Likelihood:
    \[
    \mathcal{L}(x, y_j^*) = w^T f(x, y_j^*) - \log \sum_y \exp(w^T f(x, y))
    \]
    \[
    \frac{\partial}{\partial w_i} \mathcal{L}(x, y_j^*) = f_i(x, y_j^*) - \frac{\sum_y f_i(x, y) \exp(w^T f(x, y))}{\sum_y \exp(w^T f(x, y))}
    \]
    \[
    \frac{\partial}{\partial w_i} \mathcal{L}(x, y_j) = f_i(x, y_j) - \sum_y f_i(x, y) P_w(y|x)
    \]
    \[
    \frac{\partial}{\partial w_i} \mathcal{L}(x, y_j^*) = f_i(x, y_j^*) - \mathbb{E}_y[f_i(x, y)]
    \]
    model’s expectation of feature value

- Compare to binary:
  \[
  P(y = 1|x) = \frac{\exp(w^T f(x))}{1 + \exp(w^T f(x))}
  \]
  negative class implicitly had \( f(x, y=0) \) = the zero vector

- Training: maximize
  \[
  \mathcal{L}(x, y) = \sum_{j=1}^n \log P(y_j^*|x_j)
  \]
  \[
  = \sum_{j=1}^n \left( w^T f(x_j, y_j^*) - \log \sum_y \exp(w^T f(x_j, y)) \right)
  \]
Training

\[ \frac{\partial}{\partial w_i} \mathcal{L}(x_j, y_j^*) = f_i(x_j, y_j^*) - \sum_y f_i(x_j, y) P_w(y|x_j) \]

too many drug trials, too few patients

\[ f(x, y = \text{Health}) = [1, 1, 0, 0, 0, 0, 0, 0, 0] \]
\[ f(x, y = \text{Sports}) = [0, 0, 0, 1, 1, 0, 0, 0, 0] \]

y* = Health

P_w(y|x) = [0.2, 0.5, 0.3]

(made up values)

gradient:

\[ [1, 1, 0, 0, 0, 0, 0, 0, 0] - 0.2 [1, 1, 0, 0, 0, 0, 0, 0, 0] - 0.5 [0, 0, 0, 1, 1, 0, 0, 0, 0] - 0.3 [0, 0, 0, 0, 0, 0, 1, 1, 0] \]

= [0.8, 0.8, 0, -0.5, -0.5, 0, -0.3, -0.3, 0]

Logistic Regression: Summary

- Model: \( P_w(y|x) = \frac{\exp(w^T f(x, y))}{\sum_{y' \in Y} \exp(w^T f(x, y'))} \)

- Inference: \( \arg\max_y P_w(y|x) \)

- Learning: gradient ascent on the discriminative log-likelihood

\[ f(x, y^*) - E_y[f(x, y)] = f(x, y^*) - \sum_y [P_w(y|x) f(x, y)] \]

“towards gold feature value, away from expectation of feature value”

Training

- Are all decisions equally costly?

  too many drug trials, too few patients

- We can define a loss function \( \ell(y, y^*) \)

  \[ \ell(\text{Sports}, \text{Health}) = 3 \]
  \[ \ell(\text{Science}, \text{Health}) = 1 \]

Multiclass SVM
**Multiclass SVM**

Minimize $\lambda \|w\|_2^2 + \sum_{j=1}^{m} \xi_j$

s.t. $\forall j \xi_j \geq 0$

$\forall i \ (2y_i - 1)(w^T x_i) \geq 1 - \xi_j$

$\forall j \forall y \in \mathcal{Y} \ w^T f(x_j, y_j^*) \geq w^T f(x_j, y) + \ell(y, y_j^*) - \xi_j$

Correct prediction now has to beat every other class

Score comparison is more explicit now

The 1 that was here is replaced by a loss function

**Multiclass SVM**

$\forall j \forall y \in \mathcal{Y} \ w^T f(x_j, y_j^*) \geq w^T f(x_j, y) + \ell(y, y_j^*) - \xi_j$

$w^T f(x, y) + \ell(y, y^*)$

Does gold beat every label + loss? No!

Most violated constraint is **Sports**; what is $\xi_j$?

$\xi_j = 4.3 - 2.4 = 1.9$

Perceptron would make no update here

**Computing the Subgradient**

Minimize $\lambda \|w\|_2^2 + \sum_{j=1}^{m} \xi_j$

s.t. $\forall j \xi_j \geq 0$

$\forall j \forall y \in \mathcal{Y} \ w^T f(x_j, y_j^*) \geq w^T f(x_j, y) + \ell(y, y_j^*) - \xi_j$

- One slack variable per example, so it's set to be whatever the *most violated constraint* is for that example

  $\xi_j = \max_{y \in \mathcal{Y}} \left[ w^T f(x_j, y) + \ell(y, y_j^*) - w^T f(x_j, y_j^*) \right]$

- Plug in the gold $y$ and you get 0, so slack is always nonnegative!

- If $\xi_j = 0$, the example is not a support vector, gradient is zero

- Otherwise, $\xi_j = \max_{y \in \mathcal{Y}} \left[ w^T f(x_j, y) + \ell(y, y_j^*) - w^T f(x_j, y_j^*) \right]$

  $\frac{\partial}{\partial w_i} \xi_j = f_i(x_j, y_{\text{max}}) - f_i(x_j, y_j^*)$ (update looks backwards — we're minimizing here!)

- Perceptron-like, but we update away from *loss-augmented* prediction
Putting it Together

Minimize $\lambda \| w \|^2_2 + \sum_{j=1}^{m} \xi_j$

s.t. $\forall j \xi_j \geq 0$

$\forall j \forall y \in Y \quad w^T f(x_j, y_j^*) \geq w^T f(x_j, y) + \ell(y, y_j^*) - \xi_j$

- (Unregularized) gradients:
  - SVM: $f(x, y^*) - f(x, y_{\text{max}})$ (loss-augmented max)
  - Log reg: $f(x, y^*) - E_y[f(x, y)] = f(x, y^*) - \sum_y [P_w(y|x)f(x, y)]$

- SVM: max over ys to compute gradient. LR: need to sum over ys

Structured Prediction

- Four elements of a structured machine learning method:
  - Model: probabilistic, max-margin, deep neural network
  - Objective

Objective

- Inference: just maxes and simple expectations so far, but will get harder
- Training: gradient descent?

Optimization

- Stochastic gradient *ascent*
  - Very simple to code up
  - “First-order” technique: only relies on having gradient
  - Setting step size is hard (decrease when held-out performance worsens?)

- Newton’s method
  - Second-order technique
    - Optimizes quadratic instantly
      - $w \leftarrow w + \left( \frac{\partial^2}{\partial w^2} \mathcal{L} \right)^{-1} g$
      - Inverse Hessian: $n \times n$ mat, expensive!

- Quasi-Newton methods: L-BFGS, etc. approximate inverse Hessian
### AdaGrad

- Optimized for problems with sparse features
- Per-parameter learning rate: smaller updates are made to parameters that get updated frequently
  \[ w_i \leftarrow w_i + \alpha \frac{1}{\sqrt{\epsilon + \sum_{\tau=1}^{t} g_{\tau,i}^2}} g_{\tau,i} \]  
  (smoothed) sum of squared gradients from all updates
- Generally more robust than SGD, requires less tuning of learning rate
- Other techniques for optimizing deep models — more later!

---

### Summary

- You’ve now seen everything you need to implement multi-class classification models
- Next time: HMMs (POS tagging)
- In 2 lectures: CRFs (NER)