Machine Learning

4771

Instructors:
Adrian Weller and Ilia Vovsha

Lecture 14: Text Classification and Dimensionality Reduction

- Regularized Risk Minimization
- Application to Text Classification
- Principal Component Analysis (PCA) (Duda 3.8, Bishop 12.1)

Regularized Risk Minimization

- Empirical Risk Minimization gave overfitting & underfitting
- We want to add a penalty for using too many theta values
- This gives us the Regularized Risk

$$R_{empirical} = \tfrac{1}{N}\sum_{i=1}^{N} L\left(y_i, \theta^T x_i\right)$$

$$R_{regularized} = \tfrac{1}{N}\sum_{i=1}^{N} L\left(y_i, \theta^T x_i\right) + \tfrac{\lambda}{2}\left\|\theta\right\|^2$$

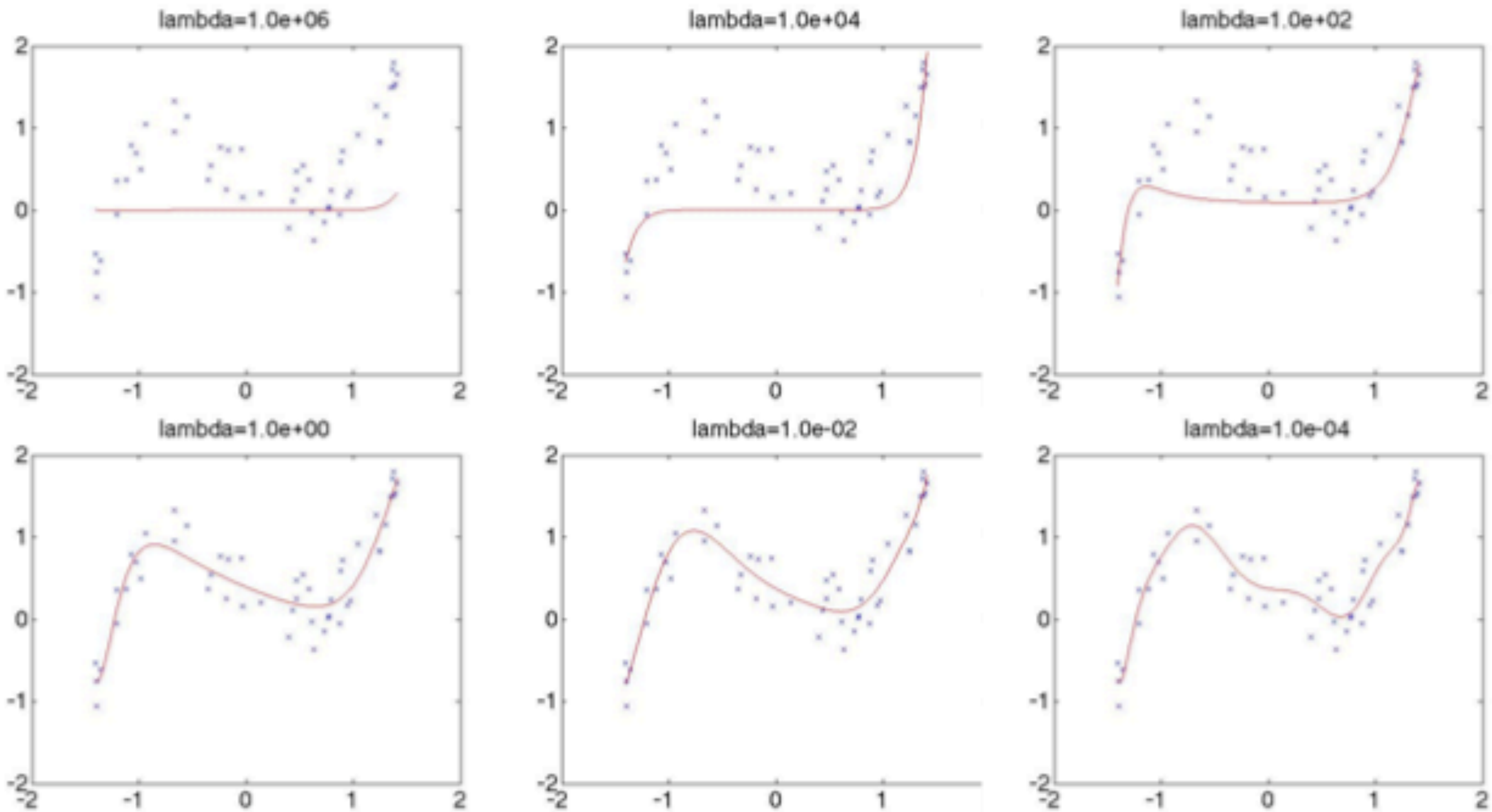
- Solution for Regularized Risk with Least Squares Loss:

$$\nabla_\theta R_{regularized} = 0 \quad \Rightarrow \quad \nabla_\theta\left(\tfrac{1}{2N}\left\|\mathbf{y} - \mathbf{X}\theta\right\|^2 + \tfrac{\lambda}{2}\left\|\theta\right\|^2\right) = 0$$

$$\theta^* = \left(\mathbf{X}^T\mathbf{X} + \lambda I\right)^{-1}\mathbf{X}^T\mathbf{y}$$

Regularized Risk Minimization

- Set P to 15 throughout. Try varying λ instead.
- Minimize $R_{regularized}(\theta)$ to get θ^*, observe $R_{empirical}(\theta^*)$



Text: Naïve Bayes

$$\in \begin{cases} \textit{religion} \\ \textit{politics} \end{cases}$$

- Text classification: simplest model
- There are about 50,000 words in English
- Each document is D=50,000 dimensional binary vector $\vec{x}_i$
- Each dimension is a word, set to 1 if word in the document

  Dim1: "the" = 1
  Dim2: "hello" = 0
  Dim3: "and" = 1
  Dim4: "happy" = 1
  ...

- Naïve Bayes: assumes each word is independent

$$p(\vec{x}) = p(\vec{x}(1),\ldots,\vec{x}(D)) = \prod_{d=1}^{D} p(\vec{x}(d))$$

$$= \prod_{d=1}^{D} \vec{\alpha}(d)^{\vec{x}(d)} (1-\vec{\alpha}(d))^{(1-\vec{x}(d))}$$

- Each 1 dimensional alpha(d) is a Bernoulli parameter
- The whole alpha vector is multivariate Bernoulli

Text: Naïve Bayes

- Maximum likelihood: assume we have several IID vectors
- Have N documents, each a 50,000 dimension binary vector
- Each dimension is a word, set to 1 if word in the document

			$\vec{x}_1$	$\vec{x}_2$	$\vec{x}_3$	$\vec{x}_4$
Dim1:	**"the"**	**=**	**1**	**0**	**1**	**1**
Dim2:	**"hello"**	**=**	**0**	**1**	**0**	**1**
Dim3:	**"and"**	**=**	**1**	**1**	**0**	**1**
Dim4:	**"happy"**	**=**	**1**	**0**	**0**	**1**

- Likelihood $= \prod_{i=1}^{N} p\left(\vec{x}_i \mid \vec{\alpha}\right) = \prod_{i=1}^{N} \prod_{d=1}^{50000} \vec{\alpha}\left(d\right)^{\vec{x}_i(d)} \left(1 - \vec{\alpha}\left(d\right)\right)^{\left(1-\vec{x}_i(d)\right)}$
- Max likelihood solution: for each word d count number of documents it appears in divided by total N documents $\vec{\alpha}\left(d\right) = \frac{N_d}{N}$
- To classify a new document x, build two models α_{+1} α_{-1} & compare $prediction = \arg\max_{y=\{\pm 1\}} p\left(\vec{x} \mid \vec{\alpha}_y\right)$

Multinomial Probability Models

- **Multinomial:** beyond binary multi-category event (dice)

1	2	3	4	5	6
$\vec{\alpha}(1)$	$\vec{\alpha}(2)$	$\vec{\alpha}(3)$	$\vec{\alpha}(4)$	$\vec{\alpha}(5)$	$\vec{\alpha}(6)$

$$p(x) = \prod_{m=1}^{M} \vec{\alpha}(m)^{\vec{x}(m)} \qquad \sum_m \vec{\alpha}(m) = 1 \qquad \vec{x} \in \mathbb{B}^M \; ; \sum_m \vec{x}(m) = 1$$

$\vec{x}(1)$	$\vec{x}(2)$	$\vec{x}(3)$	$\vec{x}(4)$	$\vec{x}(5)$	$\vec{x}(6)$

- Maximum Likelihood (IID):

$$\sum_{i=1}^{N} \log p(\vec{x}_i \mid \vec{\alpha}) = \sum_{i=1}^{N} \log \prod_{m=1}^{M} \vec{\alpha}(m)^{\vec{x}_i(m)} = \sum_{i=1}^{N} \sum_{m=1}^{M} \vec{x}_i(m) \log\left(\vec{\alpha}(m)\right)$$

- Can't just take gradient, constraint: $\sum_m \vec{\alpha}(m) - 1 = 0$
- Try using Lagrange multipliers:

$$\frac{\partial}{\partial \alpha_q} \sum_{i=1}^{N} \sum_{m=1}^{M} \vec{x}_i(m) \log\left(\vec{\alpha}(m)\right) - \lambda\left(\sum_{m=1}^{M} \vec{\alpha}(m) - 1\right) = 0$$

$$\sum_{i=1}^{N} \left(\vec{x}_i(q) \frac{1}{\vec{\alpha}(q)} \right) - \lambda = 0$$

$$\vec{\alpha}(q) = \tfrac{1}{\lambda} \sum_{i=1}^{N} \vec{x}_i(q)$$

Multinomial Probability (ML)

- Taking the gradient with Lagrangian gives this formula for each q: $\vec{\alpha}(q) = \frac{1}{\lambda}\sum_{i=1}^{N}\vec{x}_i(q)$
- Recall the constraint: $\sum_m \vec{\alpha}(m) - 1 = 0$
- Plug in α's solution: $\sum_m \frac{1}{\lambda}\sum_{i=1}^{N}\vec{x}_i(m) - 1 = 0$
- Gives the lambda: $\lambda = \sum_m \sum_{i=1}^{N}\vec{x}_i(m)$
- Final answer:

$$\vec{\alpha}(q) = \frac{\sum_{i=1}^{N}\vec{x}_i(q)}{\sum_m \sum_{i=1}^{N}\vec{x}_i(m)} = \frac{N_q}{N}$$

- Example: Rolling dice
  1,6,2,6,3,6,4,6,5,6

x=1	x=2	x=3	x=4	x=5	x=6
0.1	0.1	0.1	0.1	0.1	0.5

Text: Multinomial Counts

- **Multinomial:** can also *count many* multi-category events

  Dice: 1,3,1,4,6,1,1   Word Dice: the, dog, jumped, the

- Document i: has W_i=2000 words, each an IID dice roll

$$p(doc_i) = p\left(\vec{x}_i^1, \vec{x}_i^2, \dots, \vec{x}_i^{W_i}\right) = \prod_{w=1}^{W_i} p\left(\vec{x}_i^w\right) = \prod_{w=1}^{W_i} \prod_{d=1}^{D} \vec{\alpha}\left(d\right)^{\vec{x}_i^w(d)}$$

- Get count of each time an event occurred

$$p(doc_i) = \prod_{w=1}^{W_i} \prod_{d=1}^{D} \vec{\alpha}\left(d\right)^{\vec{x}_i^w(d)} = \prod_{d=1}^{D} \vec{\alpha}\left(d\right)^{\sum_{w=1}^{W_i} \vec{x}_i^w(d)} = \prod_{d=1}^{D} \vec{\alpha}\left(d\right)^{\vec{X}_i(d)}$$

- BUT: order shouldn't matter when "counting" so multiply by # of possible choosings. Choosing X(1),…X(D) from N

$$\binom{W_i}{\vec{X}_i(1), \dots, \vec{X}_i(D)} = \frac{W_i!}{\prod_{d=1}^{D} \vec{X}_i(d)!} = \frac{\left(\sum_{d=1}^{D} \vec{X}_i(d)\right)!}{\prod_{d=1}^{D} \vec{X}_i(d)!}$$

- Bag-of-words model (only # of words matters, not order):

$$p\left(doc_i\right) = p\left(\vec{X}_i\right) = \frac{\left(\sum_{d=1}^{D} \vec{X}_i(d)\right)!}{\prod_{d=1}^{D} \vec{X}_i(d)!} \prod_{d=1}^{D} \vec{\alpha}\left(d\right)^{\vec{X}_i(d)} \quad \sum_d \vec{\alpha}\left(d\right) = 1 \;\; X \in \mathbb{Z}_+^D$$

Text: Multinomial Counts

$$\in \begin{cases} religion \\ politics \end{cases}$$

- Text classification: bag-of-words model
- Each document is 50,000 dimensional vector
- Each dimension is a word, set to # times word in doc

			X_1	X_2	X_3	X_4
Dim1:	**"the"**	**=**	**9**	**3**	**1**	**0**
Dim2:	**"hello"**	**=**	**0**	**5**	**3**	**0**
Dim3:	**"and"**	**=**	**6**	**2**	**2**	**2**
Dim4:	**"happy"**	**=**	**2**	**5**	**1**	**0**
...						

- Each document is a vector of multinomial counts

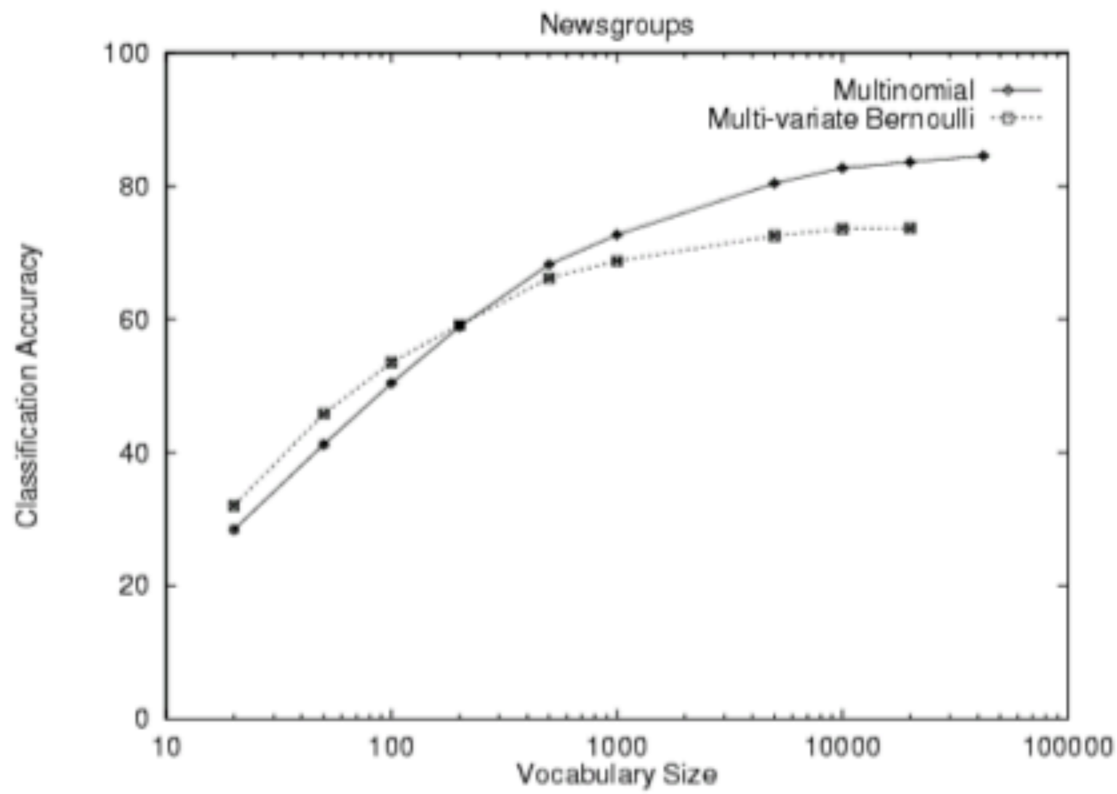
$$p\left(doc_i\right) = p\left(\vec{X}_i\right) = \frac{\left(\sum_{d=1}^{D} \vec{X}_i(d)\right)!}{\prod_{d=1}^{D} \vec{X}_i(d)!} \prod_{d=1}^{D} \vec{\alpha}(d)^{\vec{X}_i(d)} \qquad \sum_d \vec{\alpha}(d) = 1 \quad X \in \mathbb{Z}_+^D$$

- Likelihood: $l(\vec{\alpha}) = \sum_{i=1}^{N} \log p\left(\vec{X}_i\right) = \sum_{i=1}^{N} \log \frac{\left(\sum_{d=1}^{D} \vec{X}_i(d)\right)!}{\prod_{d=1}^{D} \vec{X}_i(d)!} \prod_{d=1}^{D} \vec{\alpha}(d)^{\vec{X}_i(d)}$

$$\propto \sum_{i=1}^{N} \sum_{d=1}^{D} \vec{X}_i(d) \log \vec{\alpha}(d)$$ **same formula as Multinomial ML**

Text: Models Comparison

- For text modeling (McCallum & Nigam '98)
  - Bernoulli better for small vocabulary
  - Multinomial better for large vocabulary



Dimensionality Reduction

- Problem: data might have excessive dimensionality
- Not just a computational issue! May worsen even very effective algorithms (e.g. similarity measure between examples can be adversely affected)
- Solution: reduce data dimensionality by removing (redundant) features or combining them
- Idea: project high-dimensional data onto a lower dimensional space
- How to project data? What should the projection be?
    a. Best representation of the data in some sense (Principal Component Analysis)
    b. Best separation of the data (Multiple Discriminant Analysis)

Principal Component Analysis (PCA)

- Given a set of vectors, each with dimensionality = d, we wish to project the data onto a subspace of dimensionality M < D
- Goal: maximize the variance of the projected data
- Two cases:
    1. M is given a priori
    2. We choose M based on some criteria

$$\{x_1, \ldots, x_N\}, \quad x_i \in \Re^D$$

$$\Downarrow$$

$$\{p_1, \ldots, p_N\}, \quad p_i \in \Re^M$$