Day 5: Generative models, structured classification

Introduction to Machine Learning Summer School
June 18, 2018 - June 29, 2018, Chicago

Instructor: Suriya Gunasekar, TTI Chicago

22 June 2018
Topics so far

• Linear regression
• Classification
  o nearest neighbors, decision trees, logistic regression
• Yesterday
  o Maximum margin classifiers, Kernel trick
• Today
  o Quick review of probability
  o Generative models – naive Bayes classifier
  o Structured Prediction – conditional random fields
Several slides adapted from David Sontag who in turn credits Luke Zettlemoyer, Carlos Guestrin, Dan Klein, and Vibhav Gogate
Bayesian/probabilistic learning

• Uses probability to model data and/or quantify uncertainties in prediction
  o Systematic framework to incorporate prior knowledge
  o Framework for composing and reasoning about uncertainty
  o What is the confidence in the prediction given observations so far?

• Model assumptions need not hold (and often do not hold) in reality
  o even so, many probabilistic models work really well in practice
Quick overview of random variables

• **Random variables:** A variable about which we (may) have uncertainty
  - e.g., $W = \text{weather tomorrow}$, or $T = \text{temperature}$
• For all random variables $X$ domain $\mathcal{X}$ of $X$ is the set of values $X$ can take
• **Discrete random variables:** probability distribution is a table

<table>
<thead>
<tr>
<th>$T$</th>
<th>$P$</th>
</tr>
</thead>
<tbody>
<tr>
<td>warm</td>
<td>0.5</td>
</tr>
<tr>
<td>cold</td>
<td>0.5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$W$</th>
<th>$P$</th>
</tr>
</thead>
<tbody>
<tr>
<td>sun</td>
<td>0.6</td>
</tr>
<tr>
<td>rain</td>
<td>0.1</td>
</tr>
<tr>
<td>fog</td>
<td>0.3</td>
</tr>
<tr>
<td>meteor</td>
<td>0.0</td>
</tr>
</tbody>
</table>

  - For discrete RV $X$, $\forall x \in \mathcal{X}, \Pr(X = x) \geq 0$ and $\sum_{x \in \mathcal{X}} \Pr(X = x) = 1$
• **Continuous random** $X$ with domain $\mathcal{X} \subseteq \mathbb{R}$
  - **Cumulative distribution function** $F_X(t) = \Pr(X \leq t)$
    - again $F_X(t) \in [0,1]$ and also $F_X(-\infty) = 0, F_X(+\infty) = 1$
  - **Probability density function** (if exists) $P_X(t) = \frac{dF_X(t)}{dt}$
    - Is always positive, but can be greater than 1
Quick overview of random variables

• **Expectation**

Discrete RV  \[ E[f(X)] = \sum_{x \in X} f(x) \Pr(X = x) \]

• **Mean**  \( E[X] \)

• **Variance**  \( E[(X - E[X])^2] \)
Joint distributions

- Joint distribution of random variables $X_1, X_2, ..., X_d$ is defined for all $x_1 \in X_1, x_2 \in X_2, ..., x_d \in X_d$

  $$p(x_1, x_2, ..., x_d) = \Pr(X_1 = x_1, X_2 = x_2, ..., X_d = x_d)$$

- How may numbers needed for $d$ variables each having domain of $K$ values?
  
  - $K^d$!! Too many numbers, usually some assumption is made to reduce number of probabilities
Marginal distribution

- Sub-tables obtained by elimination of variables
- Probability distribution of a subset of variables

\[
P(X_1 = x_1) = \sum_{x_2} P(X_1 = x_1, X_2 = x_2)
\]
Marginal distribution

• Sub-tables obtained by elimination of variables
• Probability distribution of a subset of variables
• Given: joint distribution
  \[ p(x_1, x_2, ..., x_d) = \Pr(X_1 = x_1, X_2 = x_2, ..., X_d = x_d) \]
  for \( x_1 \in X_1, x_2 \in X_2, ..., x_d \in X_d \)
• Say we want get a marginal of just \( x_1, x_2, x_5 \),
  that is we want to get
  \[ p(x_1, x_2, x_4) = \Pr(X_1 = x_1, X_2 = x_2, X_4 = x_4) \]
• This can be obtained by marginalizing
  \[ p(x_1, x_2, x_4) = \sum_{z_3 \in X_3} \sum_{z_5 \in X_5} ... \sum_{z_d \in X_d} p(x_1, x_2, z_3, x_4, z_5, ..., z_d) \]
Conditioning

• Random variables $X$ and $Y$ with domains $\mathcal{X}$ and $\mathcal{Y}$

$$\Pr(X = x | Y = y) = \frac{\Pr(X = x, Y = y)}{\Pr(Y = y)}$$

• Probability distributions of a subset of variables with fixed values of others
Conditioning

• Random variables $X$ and $Y$ with domains $\mathcal{X}$ and $\mathcal{Y}$

\[
\Pr(X = x | Y = y) = \frac{\Pr(X = x, Y = y)}{\Pr(Y = y)}
\]

• Conditional expectation

\[
\mathbb{E}[f(X)|Y = y] = \sum_{x \in \mathcal{X}} f(x) \Pr(X = x | Y = y)
\]

• $h(y) = \mathbb{E}[f(X)|Y = y]$ is a function of $y$

• $h(Y)$ is a random variable with distribution given by

\[
\Pr(h(Y) = h(y)) = \Pr(Y = y)
\]
Product rule

- Going from conditional distribution to joint distribution

\[
\Pr(X = x|Y = y) = \frac{\Pr(X = x, Y = y)}{\Pr(Y = y)}
\]

\[
\Pr(X = x, Y = y) = \Pr(Y = y) \Pr(X = x|Y = y)
\]

- What about the variables?

\[
\Pr(X_1 = x_1, X_2 = x_2, X_3 = x_3) = \Pr(X_1 = x_1) \Pr(X_2 = x_2|X_1 = x_1) \Pr(X_3 = x_3|X_1 = x_1, X_2 = x_2)
\]

- More generally,

\[
\Pr(X_1 = x_1, X_2 = x_2, \ldots, X_d = x_d)
\]

\[
= \Pr(X_1 = x_1) \prod_{k=2}^{d} \Pr(X_k = x_k|X_{k-1} = x_{k-1}, X_{k-2} = x_{k-2}, \ldots, X_1 = x_1)
\]
Optimal unrestricted classifier

- **C** class classification problem \( \mathcal{Y} = \{1,2, \ldots, C\} \)

- **Population distribution** Let \((x, y) \sim \mathcal{D}\)

- Consider the population 0-1 loss of classifier \(\hat{y}(x)\)

\[
L(\hat{y}) \triangleq \mathbb{E}_{x,y} [1[y \neq \hat{y}(x)]] = \mathbb{P}_{x,y}(y \neq \hat{y}(x)) = \mathbb{P}(x) \mathbb{P}(y \neq \hat{y}(x)|x)
\]

\[
L(\hat{y}|x) = \mathbb{P}(y \neq \hat{y}(x)|x)
\]

- Risk of classifier \(\hat{y}(x)\)

- \(\mathbb{P}(y \neq \hat{y}(x)|x) = 1 - \mathbb{P}(y = \hat{y}(x)|x)\)

- Optimal unrestricted classifier or Bayes optimal classifier

\[
\hat{y}^{**}(x) = \arg\max_c \mathbb{P}(y = c|x)
\]
Generative vs discriminative models

• Recall **optimal unrestricted predictor** for following cases
  
  o Regression+squared loss $\rightarrow f^{**}(x) = E[y|x]$
  
  o Classification+ 0-1 loss $\rightarrow \hat{y}^{**}(x) = \arg\max_c \Pr(y = c|x)$

• **Non-probabilistic approach**: don't deal with probabilities, just estimate $f(x)$ directly to the data.

• **Discriminative models**: Estimate/infer the conditional density $\Pr(y|x)$
  
  o Typically uses a parametric model $f_W(x)$ of $\Pr(y|x)$

• **Generative models**: Estimate the full joint probability density $\Pr(y, x)$
  
  o Normalize to find the conditional density $\Pr(y|x)$
  
  o Specify models for $\Pr(x, y)$ or $[\Pr(x|y)$ and $\Pr(y)]$
  
  o Why? In two slides!
Bayes rule

• Optimal classifier

\[ \hat{\mathbf{y}}^\ast (x) = \underset{c}{\text{argmax}} \Pr(y = c | x) \]

• Bayes rule: \( \Pr(x, y) = \Pr(y|x) \Pr(x) = \Pr(x|y) \Pr(y) \)

\[ \hat{\mathbf{y}}^\ast (x) = \underset{c}{\text{argmax}} \Pr(y = c | x) \]
\[ = \underset{c}{\text{argmax}} \frac{\Pr(x|y = c) \Pr(y = c)}{\Pr(x)} \]
\[ = \underset{c}{\text{argmax}} \frac{\Pr(x|y = c) \Pr(y = c)}{\Pr(x)} \]
Bayes rule

• Optimal classifier

\[ \hat{y}^{**}(x) = \arg \max_c \Pr(y = c | x) \]

\[ = \arg \max_c \Pr(x | y = c) \Pr(y = c) \]

• Why is this helpful?
  
  o One conditional might be tricky to model with prior knowledge but the other simple
  
  o e.g., say we want to specify a model for digit recognition

  ![Binary images](image1.png) \rightarrow digit 1

  ▪ compare specifying \( \Pr(\text{image}|\text{digit} = 1) \) vs \( \Pr(\text{digit} = 1|\text{image}) \)
Generative model for classification

\[
\arg\max_c \Pr(y = c | x) = \arg\max_c \Pr(x | y = c) \Pr(y = c)
\]

- C class classification with binary features
  \( x \in \mathbb{R}^d \) and \( y \in \{1, 2, \ldots, C\} \)
- Want to specify \( \Pr(x | y) = \Pr(x_1, x_2, \ldots, x_d | y) \)
- If each of \( x_1, x_2, \ldots, x_d \) can take one of \( K \) values. How many parameters to specify \( \Pr(x | y) \)?
  - \( C K^d \)!! Too many
Naive Bayes assumption

Specifying $\Pr(x|y) = \Pr(x_1, x_2, \ldots, x_d|y)$ requires $C K^d$

Naive Bayes assumption:
features are independent given class $y$

• e.g., for two features

$$\Pr(x_1, x_2|y) = \Pr(x_1|y) \Pr(x_2|y)$$

• more generally,

$$\Pr(x_1, x_2, \ldots, x_d|y) = \Pr(x_1|y) \Pr(x_2|y) \ldots \Pr(x_d|y) = \prod_{k=1}^d \Pr(x_k|y)$$

• number of parameters if each of $x_1, x_2, \ldots, x_d$ can take one of $K$ values?

  $\circ C K d$
Naive Bayes classifier

- Naive Bayes assumption: features are independent given class:
  \[ \Pr(x_1, x_2, ..., x_d | y) = \prod_{k=1}^{d} \Pr(x_k | y) \]
- C classes \( Y = \{1, 2, ..., C\} \) d binary feature \( X = \{0, 1\}^d \)
- Model parameters: specify from prior knowledge and/or learn from data
  - Priors \( \Pr(y = c) \) \( \Rightarrow \) #parameters \( C - 1 \)
  - Conditional probabilities \( \Pr(x_k = 1 | y = c) \) \( \Rightarrow \) #parameters \( Cd \)
    - if \( x_1, x_2, ..., x_m \) takes one of \( K \) discrete values rather than binary \( \Rightarrow \) #parameters \( (K - 1)Cd \)
    - if \( x_1, x_2, ..., x_m \) are continuous, additionally model \( \Pr(x_k | y = c) \) as some parametric distribution, like Gaussian \( \Pr(x_k | y = c) \sim \mathcal{N}(\mu_k, \sigma) \), and estimate the parameters \( (\mu_k, \sigma) \) from data
- Classifier rule:
  \[ \hat{y}_{NB}(x) = \arg\max_c \Pr(x_1, x_2, ..., x_d | y = c) \Pr(y = c) \]
  \[ = \arg\max_c \Pr(y = c) \prod_{k=1}^{d} \Pr(x_k | y = c) \]
Digit recognizer

- **Input:** pixel grids

- **Output:** a digit 0-9

Slide credit: David Sontag
What has to be learned?

$P(Y)$

$P(F_{3,1} = \text{on}|Y)$

$P(F_{5,5} = \text{on}|Y)$
MLE for parameters of NB

• Training dataset $S = \{(x^{(i)}, y^{(i)}): i = 1, 2, ..., N\}$

• Maximum likelihood estimation for naive Bayes with discrete features and labels

• Assume $S$ has iid examples
  
  - Prior: what is the probability of observing label $y$
    
    \[
    \Pr(y = c) = \frac{\sum_{i=1}^{N} 1[y^{(i)} = c]}{N}
    \]

  - Conditional distribution:
    
    \[
    \Pr(x_k = z_k|Y = c) = \frac{\sum_{i=1}^{N} 1[x_k^{(i)} = z_k, y^{(i)} = c]}{\sum_{i=1}^{N} 1[y^{(i)} = c]}
    \]
MLE for parameters of NB

- Training amounts to, for each of the classes, averaging all of the examples together:
Smoothing for parameters of NB

• Training dataset $S = \{(x^{(i)}, y^{(i)}): i = 1, 2, ..., N\}$

• Maximum likelihood estimation for naive Bayes with discrete features and labels

• Assume $S$ has iid examples
  
  o Prior: what is the probability of observing label $y$

  $$\Pr(y = c) = \frac{\sum_i 1[y^{(i)} = c]}{N}$$

  o Conditional distribution:

  $$\Pr(x_k = z_k | Y = c) = \frac{\sum_i 1[x^{(i)}_k = z_k, y^{(i)} = c]}{\sum_i 1[y^{(i)} = c]}$$
Smoothing for parameters of NB

- Training dataset $S = \{(x^{(i)}, y^{(i)}): i = 1,2, ..., N \}$
- Maximum likelihood estimation for naive Bayes with discrete features and labels
- Assume $S$ has iid examples
  - Prior: what is the probability of observing label $y$
    \[ \Pr(y = c) = \frac{\sum_i 1[y^{(i)} = c]}{N} \]
  - Conditional distribution:
    \[ \Pr(x_k = z_k | Y = c) = \frac{\sum_i 1[x_k^{(i)} = z_k, y^{(i)} = c]}{\sum_i 1[y^{(i)} = c]} + \epsilon \]
Smoothing for parameters of NB

• Training dataset \( S = \{(x^{(i)}, y^{(i)}): i = 1,2, \ldots, N\} \)

• Maximum likelihood estimation for naive Bayes with discrete features and labels

• Assume \( S \) has iid examples
  
  o Prior: what is the probability of observing label \( y \)
  
  \[
  \Pr(y = c) = \frac{\sum_i 1[y^{(i)} = c]}{N}
  \]

  o Conditional distribution:
  
  \[
  \Pr(x_k = z_k | Y = c) = \frac{\sum_i 1[x_k^{(i)} = z_k, y^{(i)} = c]}{\sum_i 1[y^{(i)} = c] + \sum_{k'} \epsilon} + \epsilon
  \]
Missing features

One of the key strengths of Bayesian approaches is that they can naturally handle missing data

- What happens if we don’t have value of some feature $x_k^{(i)}$
  - e.g., applicants credit history unknown
  - e.g., some medical tests not performed
- How to compute $\Pr(x_1, x_2, ... x_{j-1}, ?, x_{j+1} ..., x_d | y)$?
  - e.g., three coin tosses $E = \{H, ?, T\}$
  - $\Rightarrow \Pr(E) = \Pr(\{H, H, T\}) + \Pr(\{H, T, T\})$
- More generally
  \[
  \Pr(x_1, x_2, ... x_{j-1}, ?, x_{j+1} ..., x_d | y) = \sum_{z_j} \Pr(x_1, x_2, ... x_{j-1}, z_j, x_{j+1} ..., x_d | y)
  \]
Missing features in naive Bayes

\[
\Pr(x_1, x_2, \ldots x_{j-1}, ?, x_{j+1} \ldots, x_d | y) = \sum_{z_j} \Pr(x_1, x_2, \ldots x_{j-1}, z_j, x_{j+1} \ldots, x_d | y) \\
= \sum_{z_j} \left[ \Pr(z_j | y) \prod_{k \neq j} \Pr(x_k | y) \right] \\
= \prod_{k \neq j} \Pr(x_k | y) \sum_{z_j} \Pr(z_j | y) \\
= \prod_{k \neq j} \Pr(x_k | y)
\]

- Simply ignore the missing values and compute likelihood based only observed features
- no need to fill-in or explicitly model missing values
Naive Bayes

- Generative model
  - Model $P(x|y)$ and $P(y)$

- **Prediction:** models the full joint distribution and uses Bayes rule to get $P(y|x)$

- Can generate data given label
- Naturally handles missing data

Logistic Regression

- Discriminative model
  - Model $P(y|x)$

- **Prediction:** directly models what we want $P(y|x)$

- Cannot generate data
- Cannot handle missing data easily