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

TOYOTA TECHNOLOGICAL INSTITUTE AT CHICAGO

THE UNIVERSITY OF CHICAGO

NSF

Topics so far

- Linear regression
- Classification
  - nearest neighbors, decision trees, logistic regression
- Yesterday
  - Maximum margin classifiers, Kernel trick
- Today
  - Quick review of probability
  - Generative models – naive Bayes classifier
  - 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
  - Systematic framework to incorporate prior knowledge
  - Framework for composing and reasoning about uncertainity
  - What is the confidence in the prediction given observations so far?
- Model assumptions need not hold (and often do not hold) in reality
  - 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., $W = weather\ tomorrow$, or $T = temperature$
- 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

$P(T)$

T	P
warm	0.5
cold	0.5

$P(W)$

W	P
sun	0.6
rain	0.1
fog	0.3
meteor	0.0

$\Pr(W = sun) = 0.6$

  - 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) $P_X(t) = \frac{\mathrm{d}F_X(t)}{\mathrm{dt}}$
    - Is always positive, but can be greater than 1

Quick overview of random variables

- **Expectation**

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

- **Mean** $\mathbf{E}[X]$
- **Variance** $\mathbf{E}[(X - \mathbf{E}[X])^2]$

Joint distributions

- Joint distribution of random variables $X_1, X_2, \ldots, X_d$ is defined for all $x_1 \in \mathcal{X}_1, x_2 \in \mathcal{X}_2, \ldots, x_d \in \mathcal{X}_d$

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

$P(T, W)$

T	W	P
hot	sun	0.4
hot	rain	0.1
cold	sun	0.2
cold	rain	0.3

- 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

$P(T, W)$

T	W	P
hot	sun	0.4
hot	rain	0.1
cold	sun	0.2
cold	rain	0.3

$$P(t) = \sum_{w} P(t, w)$$

$P(T)$

T	P
hot	0.5
cold	0.5

$$P(w) = \sum_{t} P(t, w)$$

$P(W)$

W	P
sun	0.6
rain	0.4

$$P(X_1 = x_1) = \sum_{x_2} P(X_1 = x_1, X_2 = x_2)$$

Marginal distribution

- Sub-tables obtained by elimination of variables
- Probability distribution of a subset of variables
- Given: joint distribution
  $p(x_1, x_2, \dots, x_d) = \Pr(X_1 = x_1, X_2 = x_2, \dots, X_d = x_d)$
  for $x_1 \in \mathcal{X}_1, x_2 \in \mathcal{X}_2, \dots, x_d \in \mathcal{X}_d$
- Say we want get a marginal of just x_1, x_2, x_5,
  that is we want to get
  $$p(x_1, x_2, x_4) = \Pr\,(\mathrm{X}_1 = \mathrm{x}_1, \mathrm{X}_2 = \mathrm{x}_2, X_4 = x_4)$$
- This can be obtained by mariginalizing
  $$p(x_1, x_2, x_4) = \sum_{z_3 \in \mathcal{X}_3} \sum_{z_5 \in \mathcal{X}_5} \dots \sum_{z_d \in \mathcal{X}_d} p(x_1, x_2, z_3, x_4, z_5, \dots, z_d)$$

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 subset of variables with fixed values of others

Conditional Distributions

$P(W|T)$

$P(W|T = hot)$

W	P
sun	0.8
rain	0.2

$P(W|T = cold)$

W	P
sun	0.4
rain	0.6

Joint Distribution

$P(T, W)$

T	W	P
hot	sun	0.4
hot	rain	0.1
cold	sun	0.2
cold	rain	0.3

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

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

- $h(y) = \mathbf{E}[f(X) | Y = y]$ 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

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

$$\Updownarrow$$

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

- What about thee 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,

$$\Pr(X_1 = x_1, X_2 = x_2, \dots, X_d = x_d)$$
$$= \Pr(X_1 = x_1) \prod_{k=2}^{d} \Pr(X_k = x_k | X_{k-1} = x_{k-1}, X_{k-2} = x_{k-2}, \dots, X_1 = x_1)$$

Optimal unrestricted classifier

- C class classification problem $\mathcal{Y} = \{1,2,\ldots,C\}$
- **Population distribution** Let $(\boldsymbol{x}, y) \sim \mathcal{D}$
- Consider the population 0-1 loss of classifier $\hat{y}(x)$

$$L(\hat{y}) \triangleq \mathbf{E}_{\boldsymbol{x},y}\big[\mathbf{1}[y \neq \hat{y}(\boldsymbol{x})]\big] = \Pr_{\mathrm{x,y}}\big(y \neq \hat{y}(\boldsymbol{x})\big)$$

$$= \Pr(\boldsymbol{x}) \underbrace{\Pr(y \neq \hat{y}(\boldsymbol{x})|\boldsymbol{x})}$$

Risk of classifier $\hat{y}(\boldsymbol{x})$

Conditional risk $L(\hat{y}|\boldsymbol{x})$

- $\Pr(y \neq \hat{y}(\boldsymbol{x})|\boldsymbol{x}) = 1 - \Pr(y = \hat{y}(\boldsymbol{x})|\boldsymbol{x})$

Check that this is minimized for $\hat{y}(x) = \operatorname{argmax}_c \Pr(y = c|x)$

- Optimal unrestricted classifier or Bayes optimal classifier

$$\hat{y}^{**}(\boldsymbol{x}) = \operatorname*{argmax}_c \Pr(y = c|\boldsymbol{x})$$

Generative vs discriminative models

- Recall **optimal unrestricted predictor** for following cases
  - Regression+squared loss→ $f^{**}(\boldsymbol{x}) = \mathbf{E}[y|\boldsymbol{x}]$
  - Classification+ 0-1 loss → $\hat{y}^{**}(\boldsymbol{x}) = \underset{c}{\operatorname{argmax}} \Pr(y = c|\boldsymbol{x})$
- Non-probabilistic approach: don't deal with probabilities, just estimate $f(\boldsymbol{x})$ directly to the data.
- Discriminative models: Estimate/infer the conditional density $\Pr(y|\boldsymbol{x})$
  - Typically uses a parametric model $f_W(\boldsymbol{x})$ of $\Pr(y|\boldsymbol{x})$
- Generative models: Estimate the full joint probability density $\Pr(y, \boldsymbol{x})$
  - Normalize to find the conditional density $\Pr(y|\boldsymbol{x})$
  - Specify models for $\Pr(\boldsymbol{x}, y)$ or $[\Pr(\boldsymbol{x}|y)$ and $\Pr(y)]$
  - Why? In two slides!

Bayes rule

- Optimal classifier

$$\hat{y}^{**}(x) = \underset{c}{\operatorname{argmax}} \Pr(y = c|x)$$

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

$$\hat{y}^{**}(x) = \underset{c}{\operatorname{argmax}} \Pr(y = c|x)$$

$$= \underset{c}{\operatorname{argmax}} \frac{\Pr(x|y = c)\Pr(y = c)}{\Pr(x)}$$

$$= \underset{c}{\operatorname{argmax}} \Pr(x|y = c)\Pr(y = c)$$

Bayes rule

- Optimal classifier

$$\hat{y}^{**}(x) = \operatorname*{argmax}_c \Pr(y = c|x)$$
$$= \operatorname*{argmax}_c \Pr(x|y = c) \Pr(y = c)$$

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

Binary images

→ digit 1

  - compare specifying $\Pr(\text{image}|\text{digit} = 1)$ vs $\Pr(\text{digit} = 1|\text{image})$

Generative model for classification

$$\underset{c}{\operatorname{argmax}} \Pr(y = c|x)$$

$$= \underset{c}{\operatorname{argmax}} \Pr(x|y = c) \Pr(y = c)$$

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

Naive Bayes assumption

Specifying $\Pr(\boldsymbol{x}|y) = \Pr(x_1, x_2, \dots, x_d|y)$ 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,

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

- number of parameters if each of $x_1, x_2, \dots, x_d$ can take one of K values?
  - $C\ K\ d$

Naive Bayes classifier

- Naive Bayes assumption: features are independent given class:

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

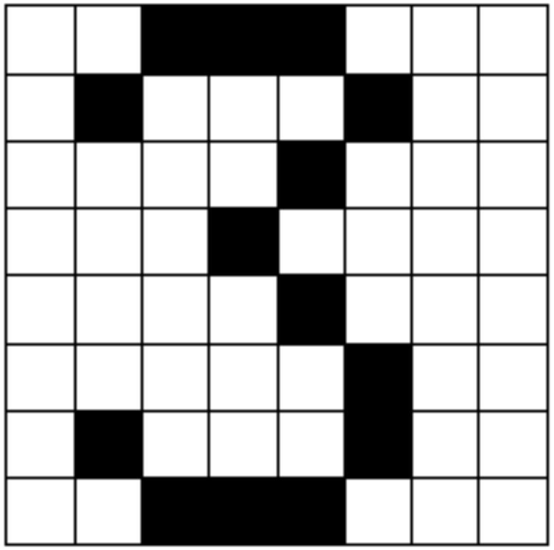
- C classes $\mathcal{Y} = \{1,2,\ldots,C\}$ d binary feature $\mathcal{X} = \{0,1\}^d$
- Model parameters: specify from prior knowledge and/or learn from data
  - Priors $\Pr(y = c)$ → #parameters $C - 1$
  - Conditional probabilities $\Pr(x_k = 1 | y = c)$ → #parameters Cd
    - if $x_1, x_2, \ldots, x_m$ takes one of K discrete values rather than binary → #parameters $(K-1)Cd$
    - if $x_1, x_2, \ldots, x_m$ are continuous, additionally model $\Pr(x_k | y = c)$ as some parametric distribution, like Gaussian $\Pr(x_k | y = c) \sim \mathcal{N}(\mu_{k,c}, \sigma)$, and estimate the parameters $(\mu_{k,c}, \sigma)$ from data
- Classifier rule:

$$\hat{y}_{NB}(x) = \underset{c}{\operatorname{argmax}} \ \Pr(x_1, x_2, \ldots, x_d | y = c) \Pr(y = c)$$

$$= \underset{c}{\operatorname{argmax}} \ \Pr(y = c) \prod_{k=1}^{d} \Pr(x_k | y = c)$$

Digit recognizer

- Input: pixel grids



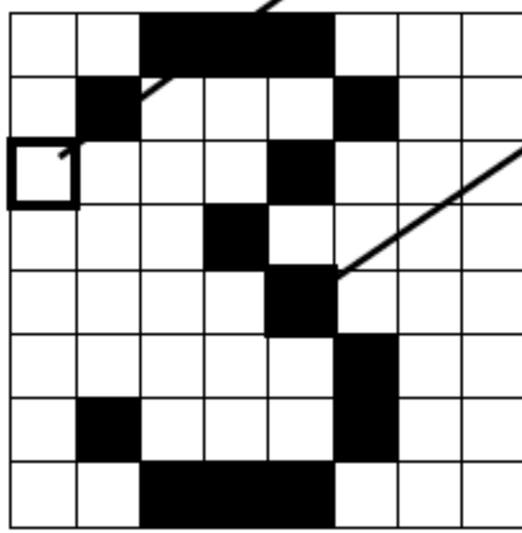
- Output: a digit 0-9

Slide credit: David Sontag

What has to be learned?

$P(Y)$

1	0.1
2	0.1
3	0.1
4	0.1
5	0.1
6	0.1
7	0.1
8	0.1
9	0.1
0	0.1



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

1	0.01
2	0.05
3	0.05
4	0.30
5	0.80
6	0.90
7	0.05
8	0.60
9	0.50
0	0.80

$P(F_{5,5} = on|Y)$

1	0.05
2	0.01
3	0.90
4	0.80
5	0.90
6	0.90
7	0.25
8	0.85
9	0.60
0	0.80

Slide credit: David Sontag

MLE 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
  - Prior: what is the probability of observing label y

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

$\sum_{c'} \sum_{i} \mathbf{1}[y^{(i)} = c']$

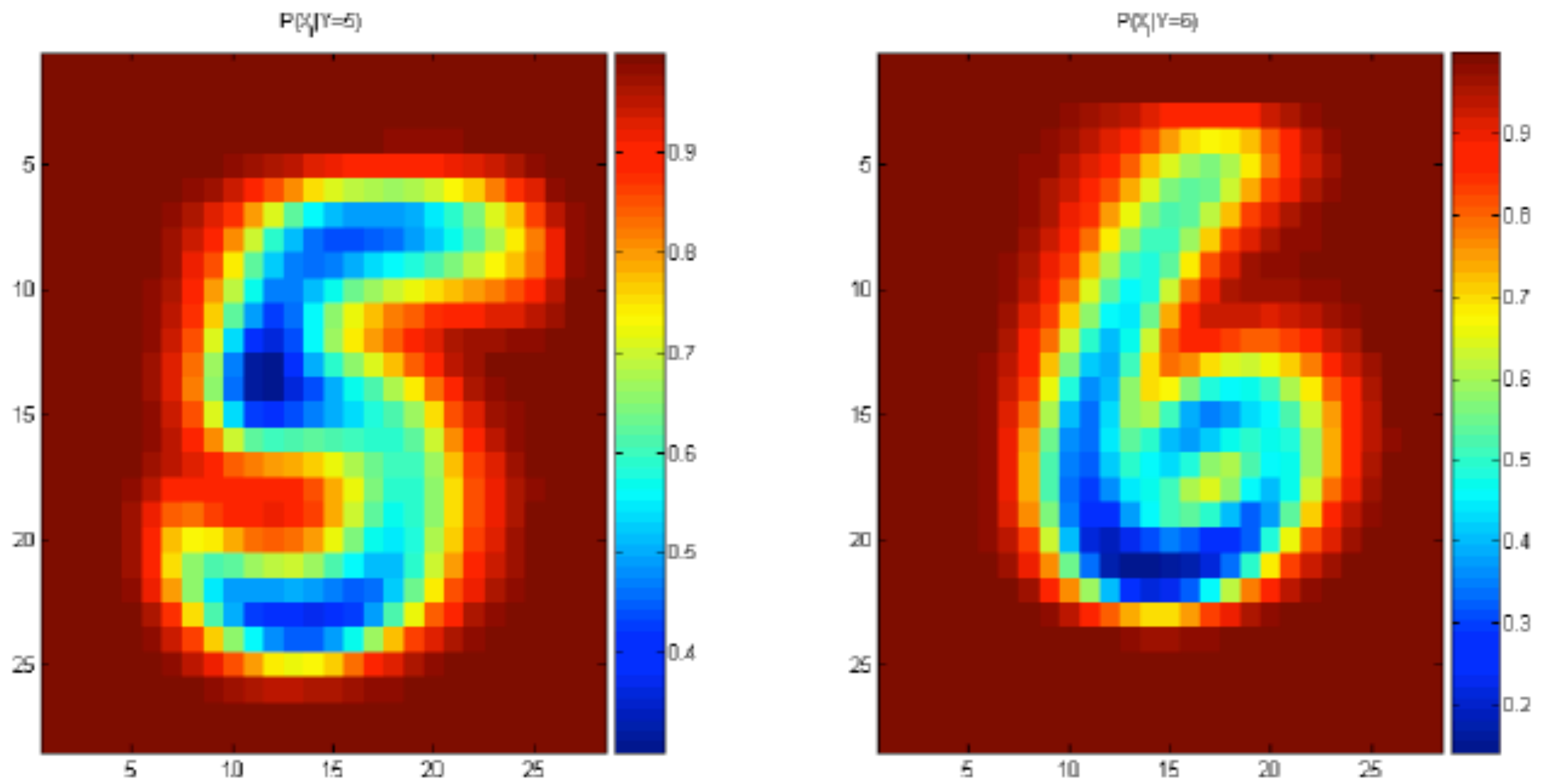
  - Conditional distribution:

$$\Pr(x_k = z_k | Y = c) = \frac{\sum_{i=1}^{N} 1[x_k^{(i)} = z_k, y^{(i)} = c]}{\sum_{i=1}^{N} 1[y^{(i)} = c]}$$

$\sum_{z_{k'}} \sum_{i} \mathbf{1}\left[x_k^{(i)} = z_{k'}, y^{(i)} = c\right]$

MLE for parameters of NB

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



Slide credit: David Sontag

Smoothing for parameters of NB

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

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

  - Conditional distribution:

$$\Pr(x_k = z_k | Y = c) = \frac{\sum_i 1\left[x_k^{(i)} = z_k, y^{(i)} = c\right]}{\sum_i 1[y^{(i)} = c]}$$

Smoothing for parameters of NB

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

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

  - Conditional distribution:

$$\Pr(x_k = z_k | Y = c) = \frac{\sum_i 1\left[x_k^{(i)} = z_k, y^{(i)} = c\right] + \epsilon}{\sum_i 1[y^{(i)} = c]}$$

Smoothing for parameters of NB

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

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

  - Conditional distribution:

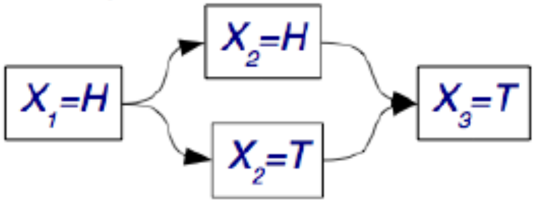
$$\Pr(x_k = z_k | Y = c) = \frac{\sum_i 1\left[x_k^{(i)} = z_k, y^{(i)} = c\right] + \epsilon}{\sum_i 1[y^{(i)} = c] + \sum_{z_{k'}} \epsilon}$$

$$\sum_{z_{k'}} \sum_i \mathbf{1}\left[x_k^{(i)} = z_{k'}, y^{(i)} = c\right] + \epsilon$$

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 $\Pr\left(x_1, x_2, \ldots x_{j-1}, ?, x_{j+1} \ldots, x_d \middle| y\right)$?
  - e.g., three coin tosses $\mathrm{E} = \{H, ?, T\}$
  - $\Rightarrow \Pr(E) = \Pr(\{H, H, T\}) + \Pr(\{H, T, T\})$



- More generally

$$\Pr\left(x_1, x_2, \ldots x_{j-1}, ?, x_{j+1} \ldots, x_d \middle| y\right) = \sum_{z_j} \Pr\left(x_1, x_2, \ldots x_{j-1}, z_j, x_{j+1} \ldots, x_d \middle| y\right)$$

Slide credit: David Sontag

Missing features in naive Bayes

$$\Pr\left(x_1, x_2, \dots x_{j-1}, ?, x_{j+1} \dots, x_d \middle| y\right)$$

$$= \sum_{z_j} \Pr\left(x_1, x_2, \dots x_{j-1}, z_j, x_{j+1} \dots, x_d \middle| y\right)$$

$$= \sum_{z_j} \left[\Pr\left(z_j \middle| y\right) \prod_{k \neq j} \Pr(x_k | y)\right]$$

$$= \prod_{k \neq j} \Pr(x_k | y) \sum_{z_j} \Pr\left(z_j \middle| y\right)$$

$$= \prod_{k \neq j} \Pr(x_k | y)$$

- 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 $\Pr(\boldsymbol{x}|y)$ and $\Pr(y)$
- Prediction: models the full joint distribution and uses Bayes rule to get $\Pr(y|\boldsymbol{x})$
- Can generate data given label
- Naturally handles missing data

Logistic Regression

- Discriminative model
  - Model $\Pr(y|\boldsymbol{x})$
- Prediction: directly models what we want $\Pr(y|\boldsymbol{x})$
- Cannot generate data
- Cannot handle missing data easily