CSE 446
Logistic Regression
Winter 2012

Dan Weld

Some slides from Carlos Guestrin, Luke Zettlemoyer

Gaussian Naïve Bayes

Sometimes Assume Variance
– is independent of Y (i.e., \( \sigma_j \)),
– or independent of \( X_i \) (i.e., \( \sigma_k \))
– or both (i.e., \( \sigma \))

\[
P(Y | X) \propto P(X | Y) P(Y)
\]

\[
P(X_i = x | Y = y_k) = \mathcal{N}(\mu_{ik}, \sigma_{ik})
\]

P(Y | X) \propto P(X | Y) P(Y)

Generative vs. Discriminative Classifiers

- Want to Learn: \( h: X \mapsto Y \)
  - \( X \) – features
  - \( Y \) – target classes
- Bayes optimal classifier – \( P(Y | X) \)
- Generative classifier, e.g., Naïve Bayes:
  - Assume some functional form for \( P(X | Y), P(Y) \)
  - Estimate parameters of \( P(X | Y), P(Y) \) directly from training data
  - Use Bayes rule to calculate \( P(Y | X) \)
  - This is a ‘generative’ model
  - Also can generate a sample of the data, \( P(X) = \sum_y P(y) P(X | y) \)
- Discriminative classifiers, e.g., Logistic Regression:
  - Assume some functional form for \( P(Y | X) \)
  - Estimate parameters of \( P(Y | X) \) directly from training data
  - This is the ‘discriminative’ model
  - Directly learn \( P(Y | X) \)
  - But cannot obtain a sample of the data, because \( P(Y) \) is not available

Univariate Linear Regression

\[
h_w(x) = w_1 x + w_0
\]

Loss(h_w) = \[ \sum_{j=1}^{n} (y_j - h_w(x_j))^2 \]

Understanding Weight Space

Univariate Linear Regression

Loss(h_w) = \[ \sum_{j=1}^{n} (y_j - (w_1 x_j + w_0))^2 \]

Understanding Weight Space
Finding Minimum Loss

$$\text{Argmin}_w \text{Loss}(h_w)$$

$$h_w(x) = w_1 x + w_0$$

$$\text{Loss}(h_w) = \sum_{j=1}^{n} (y_j - (w_1 x_j + w_0))^2$$

$$\frac{\partial}{\partial w_0} \text{Loss}(h_w) = 0$$

$$\frac{\partial}{\partial w_1} \text{Loss}(h_w) = 0$$

Unique Solution!

$$\text{Argmin}_w \text{Loss}(h_w)$$

$$h_w(x) = w_1 x + w_0$$

$$w_1 = \frac{N \sum x_j y_j - (\sum x_j)(\sum y_j)}{N \sum x_j^2 - (\sum x_j)^2}$$

$$w_0 = \frac{(\sum y_j) - w_1(\sum x_j)}{N}$$

Could also Solve Iteratively

$$\text{Argmin}_w \text{Loss}(h_w)$$

$$w = \text{any point in weight space}$$

Loop until convergence

For each $$w_i$$ in $$w$$ do

$$w_i := w_i - \alpha \frac{\partial}{\partial w_i} \text{Loss}(w)$$

Multivariate Linear Regression

$$h_w(x_j) = w_0 + \sum w_i x_{j,i} = \sum w_i x_{j,i} = w^T x_j$$

$$\text{Argmin}_w \text{Loss}(h_w)$$

Unique Solution = $$(x^T x)^{-1} x^T y$$

Problem….

Overfitting

Regularize!!

Penalize high weights

$$\text{Loss}(h_w) = \sum_{j=1}^{n} (y_j - (w_1 x_j + w_0))^2 + \lambda \sum_{i=1}^{k} |w_i|^2$$

Alternatively….

$$\text{Loss}(h_w) = \sum_{j=1}^{n} (y_j - (w_1 x_j + w_0))^2 + \lambda \sum_{i=1}^{k} |w_i|$$

Regularization

L1

L2
Back to Classification

Logistic Regression

- Learn $P(Y|X)$ directly!
- Assume a particular functional form
- Not differentiable...

Logistic Regression

Learn $P(Y|X)$ directly!

- Assume a particular functional form
- Logistic Function
- Aka Sigmoid

Logistic Function in n Dimensions

$P(Y = 1|X) = \frac{1}{1 + \exp(-z)}$

Very convenient!

$P(Y = 1|X = <X_1, ..., X_n>) = \frac{1}{1 + \exp(w_0 + \sum_{i} w_i X_i)}$

implies

$P(Y = 0|X = <X_1, ..., X_n>) = \frac{\exp(w_0 + \sum_{i} w_i X_i)}{1 + \exp(w_0 + \sum_{i} w_i X_i)}$

implies

$\ln \frac{P(Y = 0|X)}{P(Y = 1|X)} = w_0 + \sum_{i} w_i X_i$

implies

Features can be discrete or continuous!

Understanding Sigmoid Functions

$g(w_0 + \sum_{i} w_i x_i) = \frac{1}{1 + \exp(-z)}$

Very convenient!

$P(Y = 1|X = <X_1, ..., X_n>) = \frac{1}{1 + \exp(w_0 + \sum_{i} w_i X_i)}$

implies

$P(Y = 0|X = <X_1, ..., X_n>) = \frac{\exp(w_0 + \sum_{i} w_i X_i)}{1 + \exp(w_0 + \sum_{i} w_i X_i)}$

implies

$\ln \frac{P(Y = 0|X)}{P(Y = 1|X)} = w_0 + \sum_{i} w_i X_i$

implies

Features can be discrete or continuous!
Logistic regression more generally

Logistic regression in more general case, where $Y \in \{y_1, ..., y_R\}$

For $k < R$:

$$P(Y = y_k | X) = \frac{\exp(w_{0k} + \sum_{i=1}^{m} w_{ik}x_i)}{1 + \sum_{j=1}^{R-1} \exp(w_{0j} + \sum_{i=1}^{m} w_{ij}x_i)}$$

For $k = R$ (normalization, so no weights for this class)

$$P(Y = y_R | X) = \frac{1}{1 + \sum_{j=1}^{R-1} \exp(w_{0j} + \sum_{i=1}^{m} w_{ij}x_i)}$$

Features can be discrete or continuous!

---

Expressing Conditional Log Likelihood

$$l(w) \equiv \sum_j \ln P(y_j | x^j, w)$$

$$\ln P(Y=0|X,w) = -\ln(1+\exp(w_0+\sum_i w_i x_i))$$

$$\ln P(Y=1|X,w) = w_0 + \sum_i w_i x_i - \ln(1+\exp(w_0+\sum_i w_i x_i))$$

$$l(w) = \sum_j y^j \ln P(y^j | x^j, w) + (1-y^j) \ln P(y^j = 0|x^j, w)$$

1 when correct answer is 1

1 when correct answer is 0

Probability of predicting 1

Probability of predicting 0

---

Maximizing Log Likelihood

$$l(w) \equiv \ln \prod_j P(y^j | x^j, w)$$

$$= \sum_j y^j (w_0 + \sum_i w_i x_i) - \ln(1+\exp(w_0+\sum_i w_i x_i))$$

Bad news: no closed-form solution to maximize $l(w)$

Good news: $l(w)$ is concave function of $w$!

No local minima

Concave functions easy to optimize

---

Loss Functions: Likelihood vs. Conditional Likelihood

Generative (Naive Bayes) Loss function: Data likelihood

$$\ln P(D|w) = \sum_{j=1}^N \ln P(x^j, y^j | w)$$

Discriminative (Logistic Regr.) Loss funct: Conditional Data Likelihood

$$\ln P(Y | x^j, w) = \sum_{j=1}^N \ln P(y^j | x^j, w)$$

Discriminative models can’t compute $P(w|D)$!

Or, “They don’t waste effort learning $P(X)$”

Focus only on $P(Y|X)$ - all that matters for classification

---

Optimizing Concave Functions

Gradient Ascent

Conditional likelihood for Logistic Regression is concave!

Find optimum with gradient ascent

Gradient:

$$\nabla W(w) = [\frac{\partial l(w)}{\partial w_0}, \ldots, \frac{\partial l(w)}{\partial w_m}]$$

Update rule:

$$w^{(t+1)} = w^{(t)} + \eta \nabla W(w)$$

Gradient ascent is simplest of optimization approaches

e.g., Conjugate gradient ascent much better (see reading)