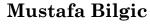
# CS578 – INTERACTIVE AND TRANSPARENT MACHINE LEARNING

**TOPIC: LOGISTIC REGRESSION** 





http://www.cs.iit.edu/~mbilgic



https://twitter.com/bilgicm

### LOGISTIC REGRESSION

- Learns P(Y|X) directly, without going through P(X|Y) and P(Y)
- Assumes P(Y|X) follows the logistic function

$$P(Y = false \mid X_1, X_2, \dots, X_n) = \frac{1}{1 + e^{w_0 + \sum_{i=1}^n w_i X_i}}$$

$$P(Y = true \mid X_1, X_2, \dots, X_n) = \frac{e^{w_0 + \sum_{i=1}^n w_i X_i}}{1 + e^{w_0 + \sum_{i=1}^n w_i X_i}}$$

• Learning: estimate the weights  $w_0, w_1, ..., w_n$ 

### Learning – Parameter Estimation

Maximize (conditional) log-likelihood

$$W \leftarrow \operatorname{argmax}_{W} \prod P(Y^{(d)}|\boldsymbol{X}^{(d)})$$

$$W \leftarrow \underset{W}{\operatorname{argmax}} \sum \ln P(Y^{(d)}|X^{(d)})$$

## TAKE DERIVATIVE OF CLL WRT W

See Lecture

4

### **OPTIMIZATION**

- No closed-form solution for W
- One solution: gradient ascent
- Good news: log-likelihood for logistic regression is concave

5

### REGULARIZATION

- Prefer smaller weights
  - Why?
- We've seen this before
  - Prefer smaller decision trees
  - Regularization for regression

# L<sub>2</sub> REGULARIZATION

Objective function

• 
$$W \leftarrow \underset{W}{\operatorname{argmax}} \left( \sum \ln P(Y^{(d)} | \boldsymbol{X}^{(d)}) - \frac{\lambda}{2} ||W||^2 \right)$$

- Trade-off between fit to the data vs model complexity
- Assuming *n* features

• 
$$W \leftarrow \underset{W}{\operatorname{argmax}} \left( \sum \ln P(Y^{(d)} | \boldsymbol{X}^{(d)}) - \frac{\lambda}{2} \sum_{i=1}^{n} w_i^2 \right)$$

• Take derivate of the objective function with respect to  $w_i$ .

## L<sub>1</sub> REGULARIZATION

- Instead of a quadratic penalty, absolute value is used
- Assuming *n* features
  - $W \leftarrow \underset{W}{\operatorname{argmax}} \left( \sum \ln P(Y^{(d)} | \boldsymbol{X}^{(d)}) \beta \sum_{i=1}^{n} |w_i| \right)$

# $L_2$ VS $L_1$

- $\circ$   $L_2$  forces the large weights to get closer to zero and places an emphasis on the large weights
  - Even though the weights get closer to zero, they are often not zero
- $\circ$   $L_1$  also penalizes large weights but the emphasis is not necessarily on the large weights
  - Some of the weights become zero
  - Leads to sparser representation
- o Can you see these?

### ALTERNATIVE FORMULATIONS

- We formulated the objective function as
  - $argmax (fit \alpha \times Complexity)$
  - Large  $\alpha$  means large penalty on complexity, i.e., smaller weights are preferred
- Alternative formulation
  - $argmin(C \times Loss + Complexity)$
  - Large *C* means large emphasis on Loss, i.e., a better fit to the data is preferred

### CATEGORICAL FEATURES

- Logistic regression's parameters are feature weights
  - Hence, features need to have values that can be multiplied by a weight
- What if you have a binary feature?
  - Two choices: 0/1, or -1/+1.
- What if you have a categorical features that has more than two possible values, such as R, G, B?
  - Incorrect way: R=1, G=2, B=3. Why?
  - How should we handle these features?

### **Z**-SCORING

- Numerical features can be readily handled by logistic regression, but a preprocessing might be a good idea
  - Otherwise, 0 is the default threshold
  - That means, for a positive weight w, anything above 0 provides positive evidence, and anything below 0 provides negative evidence (and vice versa for a negative weight w)
  - Ask yourself "is this the desired behavior for feature i my domain?"
- One approach: z-scoring
  - Subtract the mean, and divide by the standard deviation
  - See sklearn.preprocessing.StandardScaler
    - <a href="https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.Stan">https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.Stan</a> dardScaler.html

### Understanding the weights

- Similar to the interpretation of the weights of LinearRegression, Ridge, and Lasso
- A feature's importance depends on:
  - Its weight
  - The feature's variance
  - The feature's mean
  - The importance of other features

#### REFERENCES

- Tom Mitchell's freely available chapter on naïve Bayes and logistic regression
  - <a href="http://www.cs.cmu.edu/~tom/mlbook/NBayesLogReg.">http://www.cs.cmu.edu/~tom/mlbook/NBayesLogReg.</a>
    <a href="mailto:pdf">pdf</a>
- Liblinear
  - <a href="http://www.csie.ntu.edu.tw/~cjlin/papers/liblinear.pdf">http://www.csie.ntu.edu.tw/~cjlin/papers/liblinear.pdf</a>

### SCIKIT-LEARN

- http://scikitlearn.org/stable/modules/linear\_model.html#logis tic-regression
- http://scikitlearn.org/stable/modules/generated/sklearn.linea
   r\_model.LogisticRegression.html