

# Test Two Review

David Rosenberg

New York University

December 26, 2016

# Kernels

- The kernel trick

- doesn't depend on actual values of features – just ordering within each feature

# Bagging

- T/F: In bagging, as we increase the number of bootstrap samples, we expect that we will eventually overfit.
  - False.
- With bagging, how can we get an estimate of test performance while still using all our data for training?
  - “out-of-bag” error

# Random Forest

- T/F: Random forest is just bagging with trees.
  - False
- T/F: Generating too many trees in a random forest will probably lead to overfitting.
  - False

- T/F: We can use regression trees as the base classifier for AdaBoost.
  - False. AdaBoost is for hard classifiers.
- T/F: We can use SVM as a base classifier for AdaBoost.
  - True, if you map the output to  $\{-1, 1\}$  with a modified sign function.
- T/F: We can view AdaBoost a method for minimizing the exponential loss using forward stagewise additive modeling.
  - True

# Gradient Boosting

- Know how to do gradient boosting with a new loss function and a black box regression algorithm.

# Multiclass Classification

- Understand the key pieces
  - class sensitive loss function  $\Delta(y, y')$
  - feature map:  $\Psi: \mathcal{X} \times \mathcal{Y} \rightarrow \mathbf{R}^d$
  - linear score function  $(x, y) \mapsto \langle w, \Psi(x, y) \rangle$ , parameterized by  $w \in \mathbf{R}^d$
  - final prediction function  $x \mapsto \arg \max_{y \in \mathcal{Y}} \langle w, \Psi(x, y) \rangle$
  - loss functions:
    - multiclass hinge / SVM loss
    - multinomial logistic regression loss



# Likelihood, MLE, Conditional Likelihood

- go through examples in slides (Poisson regression, Gaussian regression, binomial, multinomial)
- note that you can use these same losses for gradient boosting

# Conditional Exponential Distribution

- Input:  $x$  gives location and time
- Output:  $y$  gives waiting time for taxi pickup
- Exponential distributions are a natural candidate:

$$\text{ExpDists} = \{p_\lambda(y) = \lambda e^{-\lambda y} \mathbf{1}(y \in [0, \infty)) \mid \lambda \in (0, \infty)\}.$$

- For input  $x$ , we want to give back  $\lambda$ , the exponential distribution **parameter**.
- Let's make a generalized linear model.
- So we'll predict  $x \mapsto f(w^T x)$  for some  $x$ .
- What can we use for  $f$ ?

# Conditional Exponential Distribution

- Taking  $w^T x \mapsto \exp(w^T x)$  does the trick. Maps into  $(0, \infty)$ .
- The likelihood for observation  $y \geq 0$  for

$$p_\lambda(y) = \lambda e^{-\lambda y}$$

- For input  $x$ , predicted parameter is  $\lambda = \exp(w^T x)$ .
- Likelihood of  $y | x$  is then

$$p_w(y | x) = \exp(w^T x) e^{-\exp(w^T x)y}$$

# Conditional Exponential Distribution

- Log-likelihood of  $y \mid x$  is then

$$\begin{aligned} p_w(y|x) &= \exp(w^T x) e^{-\exp(w^T x)y} \\ \implies \log p_w(y|x) &= w^T x - y \exp(w^T x) \end{aligned}$$

- Log-likelihood of  $(x_1, y_1), \dots, (x_n, y_n)$  is

$$\sum_{i=1}^n [w^T x_i - y_i \exp(w^T x_i)]$$

- MLE is then

$$\hat{w}_{\text{MLE}} = \arg \max_{w \in \mathbf{R}^d} \sum_{i=1}^n [w^T x_i - y_i \exp(w^T x_i)]$$

# Conditional Exponential Distribution with GBM?

- For linear version, we take parameter to be  $f(w^T x)$ .

$$\log p_w(y|x) = w^T x - y \exp(w^T x)$$

- Replace  $w^T x$  by a general function  $g(x)$  that we will learn with GBM.
- Log-likelihood objective function is

$$J(g) = \sum_{i=1}^n [g(x_i) - y_i \exp(g(x_i))].$$

(we want to maximize it)

- Differentiate w.r.t.  $g(x_i)$ ... etc...