# CM146 Midterm

PRASANNA SHREE KESAVA NARAYAN

TOTAL POINTS

**77 / 77**

QUESTION 1

1 Name+ID **2 / 2**

✓ **- 0 pts** **Correct**

QUESTION 2

T/F 16 pts

2.1 1-4 **8 / 8**

- **2 pts** 1 incorrect
- **2 pts** 2 incorrect
- **0 pts** 3 actually correct
- **2 pts** 4 incorrect

✓ **- 0 pts** **correct**

2.2 5-8 **8 / 8**

- **2 pts** 5 incorrect
- **2 pts** 6 incorrect
- **2 pts** 7 incorrect
- **2 pts** 8 incorrect

✓ **- 0 pts** **Correct**

QUESTION 3

Multiple Choice 32 pts

3.1 9 **4 / 4**

✓ **- 0 pts** **Correct**

- **1 pts** include d
- **1 pts** not include b
- **1 pts** include c
- **1 pts** include a

3.2 10 **4 / 4**

- **0 pts** Correct

✓ **- 0 pts** **not include b**

- **1 pts** include a

✓ **- 0 pts** **not include d**

- **1 pts** not include c

3.3 11 **4 / 4**

✓ **- 0 pts** **Correct**

- **4 pts** wrong

3.4 12 **4 / 4**

✓ **- 0 pts** **Correct**

- **1 pts** choose c
- **1 pts** not choose a
- **1 pts** not choose b
- **1 pts** choose d

3.5 13 **4 / 4**

✓ **- 0 pts** **Correct**

- **1 pts** choose b
- **1 pts** choose c
- **1 pts** not choosing d
- **1 pts** choose a

3.6 14 **4 / 4**

✓ **- 0 pts** **Correct**

- **1 pts** not choosing b
- **1 pts** not choosing c
- **1 pts** choose d
- **1 pts** not choosing a

3.7 15 **4 / 4**

✓ **- 0 pts** **Correct**

- **4 pts** Incorrect

3.8 16 **4 / 4**

✓ **- 0 pts** **Correct**

- **4 pts** Incorrect

QUESTION 4

## Decision Tree 12 pts

### 4.1 Entropy of Y 2 / 2

✓ - 0 pts Correct

- 2 pts Incorrect

### 4.2 Information Gain 6 / 6

✓ - 0 pts Correct

- 1 pts Incorrect information gain with correct X1 and X2
- 2 pts Incorrect information gain
- 4 pts Incorrect X1 or X2
- 6 pts Incorrect

### 4.3 Root choice 1 / 1

✓ - 0 pts Correct

- 1 pts Incorrect

### 4.4 Zero training error 3 / 3

✓ - 0 pts Correct

- 3 pts Improper rationale

QUESTION 5

## MLE Pareto 8 pts

### 5.1 Log likelihood 3 / 3

✓ - 0 pts Correct

- 2 pts Assumed \alpha was also raised to the power
- 1.5 pts Dropped a log
- 2 pts Assumed x_1 == x_2 == x_3...==x_n == x
- 3 pts Incorrect
- 1.5 pts Didn't raise \alpha to the Nth power
- 2 pts Didn't find likelihood of all the data
- 1 pts Miscellaneous mistake

### 5.2 Derivative 3 / 3

✓ - 0 pts Correct

- 1.5 pts Dropped the sum
- 1 pts Miscellaneous mistakes
- 1.5 pts Dropped a log
- 3 pts Incorrect

### 5.3 MLE 2 / 2

✓ - 0 pts Correct

- 1 pts Didn't solve
- 1 pts Miscellaneous mistakes
- 2 pts No answer
- 1 pts Wrong answer
- 1.5 pts Dropped the sum

QUESTION 6

## Least absolute errors 7 pts

### 6.1 Log likelihood 4 / 4

✓ - 0 pts Correct

- 2 pts log(ab)=log(a) + log(b)
- 0.5 pts 1 negative sign error
- 2 pts extra yn
- 3 pts incorrect
- 1 pts Missing term
- 4 pts no attempt

### 6.2 Equivalence 3 / 3

✓ - 0 pts Correct

- 0.5 pts slightly incorrect
- 3 pts incorrect
- 1 pts wrong steps/incorrect derivation

**4146: Introduction to Machine Learning** **Winter 2019**

Midterm

Feb. $11^{th}$, 2019

- Please do not open the exam unless you are instructed to do so.
- This is a closed book and closed notes exam.
- Everything you need in order to solve the problems is supplied in the body of this exam OR in a cheatsheet at the end of the exam.
- Mark your answers ON THE EXAM ITSELF. If you make a mess, clearly indicate your final answer (box it).
- For true/false questions, CIRCLE True OR False and provide a brief justification for full credit.
- Unless otherwise instructed, for multiple-choice questions, CIRCLE ALL CORRECT CHOICES (in some cases, there may be more than one) and provide a brief justification if the question asks for one.
- If you think something about a question is open to interpretation, feel free to ask the instructor or make a note on the exam.
- If you run out of room for your answer in the space provided, please use the blank pages at the end of the exam and indicate clearly that you've done so.
- Do NOT put answers on the back of any page of the exam.
- You may use scratch paper if needed (provided at the end of the exam).
- You have 1 hour 45 minutes.
- **Besides having the correct answer, being concise and clear is very important. For full credit, you must show your work and explain your answers.**

**Good Luck!**

Legibly write your name and UID in the space provided below to earn 2 points.

**Name:** SHREE KESAVA NARAYAN PRASANNA

**UID:** 004 973 979

| | | |
|---|---|---|
| Name and UID | | /2 |
| True/False | | /16 |
| Multiple choice | | /32 |
| Decision tree | | /12 |
| Maximum likelihood | | /8 |
| Least absolute errors | | /7 |
| **Total** | | /77 |

# True/False (16 pts)

1. (2 pts) The training error of a learning algorithm is an accurate estimate of its generalization error.

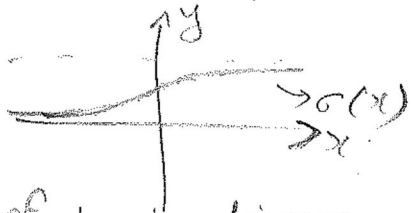
True (False)

The model may overfit, making it good for train set but not for test set.

2. (2 pts) We are trying to use logistic regression to predict disease vs healthy ($y \in \{0, 1\}$ where $y = 1$ refers to the disease class) by measuring how many steps a person walked in a week: $x \in \mathbb{R}$. From the training data, we learn a weight $w = -1$ for the feature $x$ and an intercept $b = 1$. This model will predict that the probability of disease increases as a person walks more steps.

True (False)

$h_\theta(x) = \sigma(\theta^T x) = \sigma(wx + b) = \sigma(-x + 1)$ *




As $x \to \infty$, i.e., person walks more, $\sigma(-x+1) \to 0$

∴ As they walk more, the probability of having disease decreases, as $h_\theta(x)$ is prob. that they have a disease.

3. (2 pts) For a binary classification problem, we use the perceptron algorithm to learn weights $\boldsymbol{w}$ on a training dataset (assume the intercept $b = 0$). The error of the perceptron on a test dataset is unchanged if we then rescale the weights $\boldsymbol{w}$ so that they sum to 1.

(True) False

Assuming that the direction in which $w$ points stays the same, rescaling the vector $w$ will not change results as the hyperplane ⊥ to $w$ stays the same (orthogonal).

4. (2 pts) The function $f(x) = x^3$ is convex over the set of all real numbers.

True (False)

$f''(x) = 6x$

$6x < 0 \quad \forall \quad x < 0$

∴ $f(x)$ is convex iff $f''(x) \geq 0 \quad \forall \quad x \in \mathbb{R}$

∴ $x^3$ is not convex

5. (2 pts) To predict $y$ from $\boldsymbol{x}$ where $y \in \{0,1\}, \boldsymbol{x} \in \mathbb{R}^D$, we transform $\boldsymbol{x}$ by a function $\boldsymbol{\phi}(\boldsymbol{x}) = \boldsymbol{C}\boldsymbol{x} + \boldsymbol{d}$ where $\boldsymbol{C}$ is a $M \times D$ matrix and $\boldsymbol{d} \in \mathbb{R}^M$. A logistic regression model with $\boldsymbol{\phi}(\boldsymbol{x})$ as input can learn a non-linear decision boundary.

True (False)

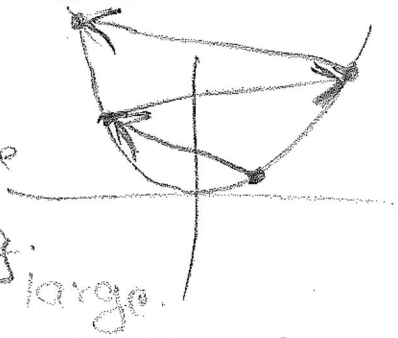
The decision boundary is still linear in the input space ($\mathbb{R}^M$) as the hypothesis is $h_\theta(\phi x) = \sigma(\theta^T \phi(x))$ and $\theta$ will be learnt. These $\theta$ define a linear dec. bound $\theta^T z = 0$, $z \in \mathbb{R}^M$. Furthermore $Cx+d$ is a linear mapping, so linear hyperplane in $\mathbb{R}^M$ is mapped to linear in $\mathbb{R}^D$.

6. (2 pts) The K-nearest neighbor algorithm often uses Euclidean distance as the default distance metric: $d(\boldsymbol{x}_i, \boldsymbol{x}_j) = \|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2$. Suppose we instead use a new distance metric: $d(\boldsymbol{x}_i, \boldsymbol{x}_j) = \log(1 + \|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2)$. The classification results will change as a result of this distance metric.

True (False)

$\log(1 + \|x_i - x_j\|_2)$ is monotonously increasing, so comparisons between $\|x_i - x_j\|_2$'s will yield same results as between $\log(1 + \|x_i - x_j\|_2)$'s.

7. (2 pts) We have a convex function for which we would like to find the minimum. For any choice of step size, gradient descent applied to our problem is always guaranteed to converge to the minimum.



True (False)

The gradient may be too large and so may keep overshooting. This happens when step size is too large.

8. (2 pts) Stochastic gradient descent is guaranteed to converge to the minimum of a convex function faster than batch gradient descent.

True (False)

· Though computation cost is decreased, stochastic GD only looks at indiv. data point's contribution to Gradient, which is not accurate to the true (Batch) gradient at that point, so may end up moving in the

# Multiple choice (32 pts)

CIRCLE ALL CORRECT CHOICES (in some cases, there may be more than one)

9. (3 pts) You are given a training dataset for a binary classification problem: $\{(x_1, y_1), \ldots, (x_N, y_N)\}$ where $(x_n, y_n), n \in \{1, \ldots, N\}$ is an instance-label pair. You use a learning algorithm to build a binary classifier $D_1$. You then take one of the instance-label pairs, add a copy of it to the training set (so your new training set will have $N + 1$ instances), and rerun the same learning algorithm to create $D_2$ (assume any random numbers used in each case are the same). $D_1$ and $D_2$ will always have identical decision boundaries when the learning algorithm used is:

   (a) Decision Tree
   (b) 1-Nearest Neighbor
   (c) 3-Nearest Neighbor
   (d) Perceptron

10. (4 pts) We want to deploy a machine learning algorithm on a website to predict what ad to show to a visitor to the website based on their view history. To learn this model, we have a training dataset of a million instances and 10 features. Which of the following learning models would be practical for this application ?

   (a) K-Nearest Neighbors
   (b) Logistic regression
   (c) Decision tree
   (d) Perceptron

11. (4 pts) Which of the following phenomenon is called over-fitting ?

   (a) low training error, low test error
   (b) low training error, high test error
   (c) high training error, low test error
   (d) high training error, high test error

12. (4 pts) In which of these settings is over-fitting more likely?

(a) Increasing the number of features

(b) Increasing the complexity of the hypothesis space

(c) Increasing the value of the regularization hyperparameter

(d) Increasing the number of training examples

13. (4 pts) Which of the following is true of the Perceptron classifier ?

(a) If the Perceptron learning algorithm finds a hypothesis that achieves zero training error, this hypothesis also achieves zero test error.

(b) If the Perceptron learning algorithm does not converge after MaxIter iterations, the problem is not linearly separable.

(c) Computational cost of classifying a test instance increases with the size of the training data.

(d) Computational cost of classifying a test instance increases with the number of features.

14. (4 pts) Which of the following methods can achieve zero training error on any linearly separable dataset?

(a) Perceptron

(b) Logistic regression

(c) 1-Nearest Neighbor

(d) 3-Nearest Neighbor

15. (4 pts) The entropy of a distribution over a set of 4 items with probability mass function $p$ is defined as $-\sum_{k=1}^{4} p(k) \log_2 p(k)$. Which of the following distributions has the largest entropy?

(a) $(0, 1, 0, 0)$

(b) $(1, 0, 0, 0)$

(c) $(0.25, 0.25, 0.25, 0.25)$

(d) $(0.5, 0.2, 0.2, 0.1)$

16. (4 pts) Let $\boldsymbol{X} \in \mathbb{R}^{N \times D}$ be the design matrix with each row corresponding to the features of an example and $\boldsymbol{y} \in \mathbb{R}^N$ be a vector of all the labels. The OLS solution is $\boldsymbol{\theta}_{OLD} = (\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X})^{-1}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{y}$. Which of the following is the OLS solution $\boldsymbol{\theta}_{NEW}$ if we scale each feature by 2 (*i.e.*, the new dataset is $2\boldsymbol{X}$)?

(a) $2\boldsymbol{\theta}_{OLD}$

(b) $4\boldsymbol{\theta}_{OLD}$

(c) $\frac{1}{2}\boldsymbol{\theta}_{OLD}$

(d) $\boldsymbol{\theta}_{OLD}$

# Decision Tree (12 pts)

**Decision tree learning**
Given the following set of training observations with two features $(X_1, X_2)$ and the response variable $Y$, we would like to learn a decision tree using information gain to choose nodes. Recall that the information gain is defined as $Gain = H[Y] - H[Y|X]$, where $H[Y] = -E[\log_2 P(Y)]$ is the entropy. See cheatsheet at the end of this exam for entropy values.

| $X_1$ | $X_2$ | $Y$ |
|---|---|---|
| 0 | 1 | 1 |
| 1 | 1 | 0 |
| 0 | 0 | 1 |
| 1 | 0 | 1 |
| 0 | 1 | 0 |
| 0 | 0 | 0 |

(a) (2 pts) What is the entropy of $Y$?

$H[Y] = -\sum_{K=0}^{1} P(Y=K) \log(P(Y=K))$

$P(Y=0) = \frac{\#\text{ of examples where } Y=0}{\text{Total \# of examples}} = \frac{3}{6} = \frac{1}{2}$

similarly, $P(Y=1) = 1/2$

$\therefore H[Y] = -\left(\frac{1}{2}\log\frac{1}{2} + \frac{1}{2}\log\frac{1}{2}\right) = -\left(\log\frac{1}{2}\right) = \log_2 2 = 1$

$\therefore H[Y] = 1$

(b) (6 pts) What is the information gain of each of the attributes $X_1$ and $X_2$ relative to $Y$?

$H[Y|X_1] = P(X_1=1)\,H[Y|X_1=1] + P(X_1=0)\,H[Y|X_1=0]$

$P(X_1=1) = 2/6 = 1/3$, $P(X_1=0) = 2/3$

$H[Y|X_1=1] = -(P(Y=1|X_1=1)\log(P(Y=1|X_1=1)) + P(Y=0|X_1=1)\log P(Y=0|X_1=1))$

$= -(1/2 \log 1/2 + 1/2 \log 1/2) = 1$

Similarly

$H[Y|X_1=0] = -(1/2 \log 1/2 + 1/2 \log 1/2) = 1$

$\therefore H[Y|X_1] = \frac{1}{3}(1) + \frac{2}{3}(1) = 1$

Info gain $= H[Y] - H[Y|X_1] = 1 - 1 = 0$

Now, for $X_2$:

$H[Y|X_2] = P(X_2=1)\,H[Y|X_2=1] + P(X_2=0)\,H[Y|X_2=0]$

$P(X_2=1) = 1/2$, $P(X_2=0) = 1/2$,

$H[Y|X_2=1] = -(P(Y=1|X_2=1)\log(P(\cdot)) + P(Y=0|X_2=\phi)\log(P(\cdot)))$

$= -(1/3 \log 1/3 + 2/3 \log 2/3) = 0.92$

$H[Y|X_2=0] = -(2/3 \log 2/3 + 1/3 \log 1/3) = 0.92$

$\therefore H[Y|X_2] = \frac{1}{2}(0.92) + \frac{1}{2}(0.92) = 0.92$

$\therefore$ Info gain $= H[Y] - H[Y|X_2] = 1 - 0.92 = 0.08$

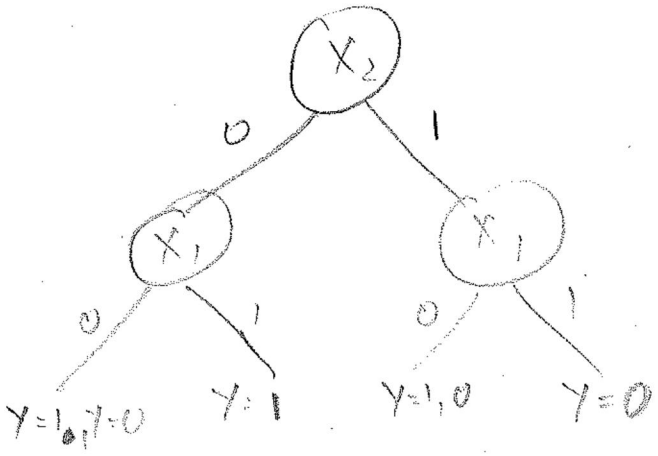
(c) (1 pts) Using information gain, which attribute will the ID3 decision tree learning algorithm choose as the root?

The algorithm would choose $X_2$ as the root (as it provides the most information gain).

(d) (3 pts) Can we construct a decision tree with zero training error on this training data? If yes, provide an example. If no, justify.

No we cannot

After splitting on $X_2$ :




We see that these are leave where it is still not known whether Y is 0 or 1.

In the case the model would make an arbitrary decision (equal # of 0's & 1's in these leaves), and so would misclassify at least 1 of the training examples.

Even if we used $X_1$ as root feature, we would not be able to get 0 train error, for similar reasons.

Intuitively, the features together do not provide enough information gain (loss in entropy or uncertainty) and so, the model will not classify all training examples correctly.

## Maximum Likelihood (8 pts)

Let $X_1, \ldots, X_N$ be i.i.d. random variables where $X_n \sim Pareto(\alpha), n \in \{1, \ldots, N\}$. The probability density function for $X \sim Pareto(\alpha)$ is:

$$f(x;\alpha) = \begin{cases} \alpha x^{-(\alpha+1)}, & \text{if } x \geq 1 \\ 0, & \text{if } x < 1 \end{cases}$$

(a) (3 pts) Give an expression for the log likelihood $l(\alpha)$ as a function of $\alpha$ given this specific dataset. You may assume all values, $X_1, \ldots, X_N \geq 1$.

$l(\alpha) = \log\left(f(X_1, X_2 \ldots, X_N; \alpha)\right)$

As $X_1 \ldots X_N$ are iid, $f(X_1, \ldots, X_N; \alpha) = \prod_{i=1}^{N} f(X_i; \alpha)$ (as they are independent)

$\therefore \log\left(\prod_{i=1}^{N} f(x_i;\alpha)\right) = \sum_{i=1}^{N} \log f(x_i;\alpha)$

Now $\log f(x_i;\alpha) = \log\left(\alpha x_i^{-(\alpha+1)}\right)$ (assuming all $x_i \geq 1$)

$= -(\alpha+1)\log(x_i) + \log\alpha$

$\therefore l(\alpha) = \sum_{i=1}^{N}\left[-(\alpha+1)\log(x_i) + \log\alpha\right] = N\log\alpha - (\alpha+1)\sum_{i=1}^{N}\log(x_i)$

$\therefore l(\alpha) = N\log\alpha - (\alpha+1)\sum_{i=1}^{N}\log(x_i)$

(b) (3 pts) Compute the derivative of the log likelihood for this specific dataset.

$$\frac{d\ell(\alpha)}{d\alpha} = \frac{N}{\alpha} - \sum_{i=1}^{N} \log(x_i)$$

(Note that $\sum_{i=1}^{N} \log(x_i)$ is constant, i.e. independent of $\alpha$)

$$\left( \frac{d\ell(\alpha)}{d\alpha} = \frac{d\left(N \log \alpha - (\alpha+1)\sum_{i=1}^{N} \log(x_i)\right)}{d\alpha} \right.$$

$$\left. = N \frac{d \log \alpha}{d\alpha} - \frac{d(\alpha \sum_{i=1}^{N} \log(x_i))}{d\alpha} - \frac{d\left(\sum_{i=1}^{N} \log(x_i)\right)}{d\alpha}_{=0} \right)$$

(c) (2 pts) What is the maximum likelihood estimate $\hat{\alpha}$ of $\alpha$?

Set $\ell'(\alpha) = 0$

$$\Rightarrow \frac{N}{\hat{\alpha}} = \sum_{i=1}^{N} \log(x_i)$$

$$\Rightarrow \hat{\alpha} = \frac{N}{\sum_{i=1}^{N} \log(x_i)}$$

(Also Note the $\ell''(\alpha) = -\frac{N}{\alpha^2} < 0 \quad \forall \alpha$

$\therefore \ell(\alpha)$ is convex, so any stationary point is minimum)

$\therefore \hat{\alpha} = \frac{N}{\sum_{i=1}^{N} \log(x_i)}$ is the MLE of $\alpha$

## Least absolute errors (7 pts)

In class, we showed how linear regression (ordinary least squares) can be interpreted as a probabilistic model. In this problem, we consider an alternative model for regression. We have a training set $\{(\boldsymbol{x}_n, y_n)\}_{n=1}^N$ where $\boldsymbol{x}_n \in \mathbb{R}^D$ and $y_n \in \mathbb{R}$. Now we model our target $y_n$ as distributed according to the following distribution:

$$p(y_n|\boldsymbol{x}_n; \boldsymbol{\theta}) = \frac{1}{2b} \exp\left(-\frac{1}{b}|y_n - \boldsymbol{\theta}^T \boldsymbol{x}_n|\right)$$

Here $|z|$ is the absolute value of $z$ and $b > 0$ is a constant that is assumed to be known.

(a) (4 pts) Write the log likelihood of the parameters $l(\boldsymbol{\theta})$. Express your answer in terms of $y_n, \boldsymbol{x}_n, \boldsymbol{\theta}$.

$$L(\theta) = P(y_1, y_2, \ldots, y_N | x_1, \ldots, x_N; \theta)$$

$$= \prod_{n=1}^{N} P(y_n | x_n; \theta) \quad \text{(as the samples are drawn independently)}$$

$$\therefore l(\theta) = \log L(\theta) = \log\left(\prod_{n=1}^{N} P(y_n | x_n; \theta)\right)$$

$$= \sum_{n=1}^{N} \log\left(P(y_n | x_n; \theta)\right)$$

$$= \sum_{n=1}^{N} \log\left(\frac{1}{2b} \exp\left(-\frac{1}{b}|y_n - \theta^T x_n|\right)\right)$$

$$= \sum_{n=1}^{N} \left(\log\left(\frac{1}{2b}\right) - \frac{1}{b}|y_n - \theta^T x_n|\right) = -N \log 2b$$

$$- \frac{1}{b} \sum_{n=1}^{N} |y_n - \theta^T x_n|$$

$$\therefore l(\theta) = -N \log 2b - \frac{1}{b} \sum_{n=1}^{N} |y_n - \theta^T x_n|$$

(b) (3 pts) Show that finding the maximum likelihood estimate of $\boldsymbol{\theta}$ leads to the same answer as finding the $\boldsymbol{\theta}$ that minimizes the cost function (which is the sum of absolute errors):

$$J(\boldsymbol{\theta}) = \sum_{n=1}^{N} |y_n - \boldsymbol{\theta}^{\mathrm{T}} \boldsymbol{x}_n|$$

∴ To maximize $L(\theta)$, we can maximize $l(\theta)$ ($\log(\,)$ is monotonously increasing) which is the same as minimizing $-l(\theta)$

∴ To find MLE of $\theta$, we can minimize:

$$-l(\theta) = N \log 2b + \frac{1}{b} \sum_{n=1}^{N} |y_n - \theta^T x_n|$$

Now, it is assumed that $b$ is known constant

∴ $N \log 2b$ is a constant, and so is $\frac{1}{b}$

∴ To minimize $-l(\theta)$, we must minimize $\frac{1}{b} \sum_{n=1}^{N} |y_n - \theta^T x_n|$, and so must minimize

$$J(\theta) = \sum_{n=1}^{N} |y_n - \theta^T x_n|$$

($\frac{1}{b}$ is a positive constant, so minimizing $\frac{1}{b} \sum_{n=1}^{N} |y_n - \theta^T x_n|$ is the same as minimizing $\sum_{n=1}^{N} |y_n - \theta^T x_n| = J(\theta)$)

# Identities

## Probability density/mass functions for some distributions

$$\text{Normal} \quad : \quad P(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

$$\text{Multinomial} \quad : \quad P(\boldsymbol{x}; \boldsymbol{\pi}) = \prod_{k=1}^{K} {\pi_k}^{x_k}$$

$\boldsymbol{x}$ is a length $K$ vector with exactly one entry equal to 1 and all other entries equal to 0

$$\text{Poisson} \quad : \quad P(x; \lambda) = \frac{\lambda^x \exp(-\lambda)}{x!}$$

## Matrix calculus

Here $\boldsymbol{x} \in \mathbb{R}^n, \boldsymbol{b} \in \mathbb{R}^n, \boldsymbol{A} \in \mathbb{R}^{n \times n}$. $\boldsymbol{A}$ is symmetric.

$$\begin{aligned} \nabla \boldsymbol{x}^{\mathrm{T}} \boldsymbol{A} \boldsymbol{x} &= 2\boldsymbol{A}\boldsymbol{x} \\ \nabla \boldsymbol{b}^{\mathrm{T}} \boldsymbol{x} &= \boldsymbol{b} \end{aligned}$$

## Entropy

The entropy $H(X)$ of a Bernoulli random variable $X \sim \mathit{Bernoulli}(p)$ for different values of $p$:

| $p$ | $H(X)$ |
|---|---|
| $\frac{1}{2}$ | 1 |
| $\frac{1}{3}$ | 0.92 |
| $\frac{1}{4}$ | 0.81 |
| $\frac{1}{5}$ | 0.73 |
| $\frac{2}{5}$ | 0.97 |

You may use this page for scratch space.

You may use this page for scratch space.