Machine Learning for Sensory Signals
Homework # 1

1. Show that
(a) ∂(x Ax) = (A + A T)x

Machine Learning Sensory Signals

Homework # 1

1. Show that

(a) $\frac{\partial}{\partial x} x^T A x = (A + A^T) x$

Proof :- let x be a $n \times 1$, and A be $n \times n$,

and α be the scalar resulting from product of

$$\alpha = x^T A x$$

and it's given by.

$$\alpha = \sum_{j=1}^{n} \sum_{i=1}^{n} a_{ij} x_i x_j$$

Differentiating wrt k^{th} element of "x" we ha

$$\frac{\partial \alpha}{\partial x_k} = \sum_{j=1}^{n} a_{kj} x_j + \sum_{i=1}^{n} a_{ij} x_j$$

for all $k = 1, 2 \dots n$, and consequently.

$$\frac{\partial \alpha}{\partial x} = x^T A^T + x^T A = x^T (A^T + A)$$

(b)∂tr(AB) = B T∂A

(b) $\frac{\partial}{\partial A} tr(AB) = B^T$

Proof:- $tr\, AB = tr \begin{bmatrix} \leftarrow \vec{a}_1 \rightarrow \\ \leftarrow \vec{a}_2 \rightarrow \\ \vdots \\ \leftarrow \vec{a}_n \rightarrow \end{bmatrix} \begin{bmatrix} \uparrow & \uparrow & \uparrow & & \uparrow \\ \vec{b}_1 & \vec{b}_2 & b_3 & \cdots & \vec{b}_n \\ \downarrow & \downarrow & & & \downarrow \end{bmatrix}$

$= tr \begin{bmatrix} \vec{a}_1^T \vec{b}_1 & \vec{a}_1^T \vec{b}_2 & \cdots & \vec{a}_1^T \vec{b}_n \\ \vec{a}_2^T \vec{b}_1 & \vec{a}_2^T \vec{b}_2 & \cdots & \vec{a}_2^T \vec{b}_n \\ \vdots & & \ddots & \vdots \\ \vec{a}_n^T b_1 & \cdots & \cdots & \vec{a}_n^T \vec{b}_n \end{bmatrix}$

$= \sum_{i=1}^{m} a_{1i} b_{i1} + \sum_{i=1}^{m} a_{2i} b_{i2} + \cdots \sum_{i=1}^{m} a_{ni} b_{in}$

$\frac{\partial tr AB}{\partial A} \Rightarrow \frac{\partial tr AB}{\partial a_{ij}}$

$= b_{ji}$

$= B^T$

Hence Proved.

2. Show that for the regression problem, the mean square error based estimate is the conditional expectation.

Q=2) Show that for the regression problem, mean squared error based estimate is the conditional expectation.

Solution:
optimal Solution is the conditional expectation,
Expanding the square term below

$$\{y(x) - t\}^2$$

$$= \{y(x) - E[t|x] + E[t|x] - t\}^2$$

$$= \{y(x) - E[t|x]\}^2 + 2\{y(x) - E[t|x]\}\{E[t|x] - t\}$$

$$+ \{E[t|x] - t\}^2$$

using $E[t|x]$ to denote $E_t[t|x]$, substituting it to the Loss function and performing integral over t.

$$E[L] = \int \{y(x) - E[t|x]\}^2 p(x)\,dx \; + $$

$$\int \{E[t|x] - t\}^2 p(x)\,dx$$

function $y(x)$ will be minimized when $y(x)$ is equal to $E[t|x]$, This shows that the optimal least square predictor is given by the conditional mean.

3. What are the different approaches to Machine learning in terms of classification settings. Enumerate the difference between Generative modeling and discriminative modeling.

Ans =

Different approaches to Machine learning in terms of classification settings:

1. Generative Modeling
   ->Estimating the likelihood and prior density seperately

2. Discriminative Modeling
   -> Directly Modeling the class posterior probability

3. Discriminant Function
   -> No probabilistic interpretation
   -> Fitting the cost function

Difference between generative and Discriminative Modelling:

1. A generative model learns the joint probability distribution p(x,y) and a discriminative model learns the conditional probability distribution p(y|x), read as "the probability of y given x".

2. The distribution p(y|x) is the natural distribution for classifying a given example x into a class y, which is why algorithms that model this directly are called discriminative algorithms.

Whereas in Generative algorithms model p(x,y), which can be tranformed into p(y|x) by applying Bayes rule and then used for classification. However, the distribution p(x,y) can also be used for other purposes. For example you could use p(x,y) to generate likely (x,y) pairs, generation of new data.

3. **Example:**
A generative algorithm models how the data was generated in order to categorize a signal. It asks the question: based on my generation assumptions, which category is most likely to generate this signal?

A discriminative algorithm does not care about how the data was generated, it simply categorizes a given signal.