Question 1 [22 pts] Logistic regression.

a) **[6 pts]** Suppose that you trained a logistic regression model on some training data and the resulting weights are $w_0 = 1, w_1 = 2, w_2 = -3$. Assuming two classes (+ and -), a data point is predicted to belong to class + when $\sigma(\boldsymbol{w}^T\boldsymbol{x} + w_0) \geq 0.5$. Classify the following data points:

i) $(1,2)^T$ $\quad (1 \quad 2 \quad -3)\begin{pmatrix}1\\1\\2\end{pmatrix} = -3 \quad \Rightarrow \quad -$ **(2pts)**

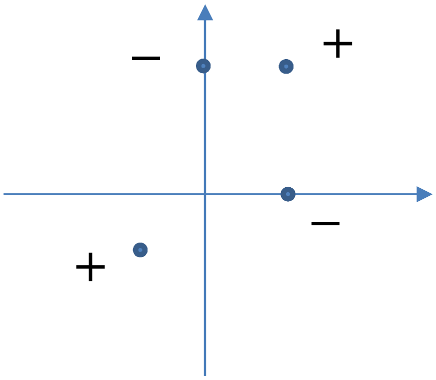
ii) $(2,1)^T$ $\quad (1 \quad 2 \quad -3)\begin{pmatrix}1\\2\\1\end{pmatrix} = 2 \quad \Rightarrow \quad +$ **(2pts)**

iii) $(0.5,-1)^T$ $\quad (1 \quad 2 \quad -3)\begin{pmatrix}1\\0.5\\-1\end{pmatrix} = 5 \quad \Rightarrow \quad +$ **(2pts)**

b) **[8 pts]** Consider the following data set:

$$\{(1,2)^T+, (-1,-1)^T+, (0,2)^T-, (1,0)^T-\}$$

where the first two points belong to class + and the last two points belong to class -. Is it possible for a logistic regression classifier to correctly classify all points in this dataset? If yes, give weights that ensure correct classification? If no, explain why and describe an approach that could be used to modify the logistic regression classifier to correctly classify all those data points?



No **(2pts)**, the dataset is not linearly separable **(2pts)**.

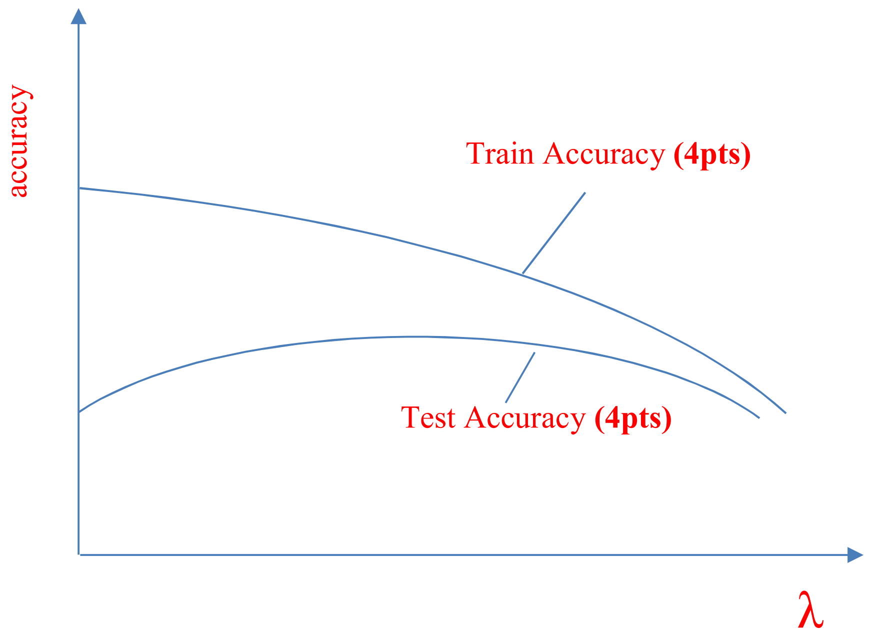
Define a non-linear mapping ϕ that maps the data into a new space that is linearly separable **(3pts)**.

Question 1 (continued)

c) **[8 pts]** Consider regularized maximum likelihood as the training objective for logistic regression:

$$\min_{\boldsymbol{w}} -\sum_{n} y_n \ln \sigma(\boldsymbol{w}^T \overline{\boldsymbol{x}}_n) + (1 - y_n) \ln(1 - \sigma(\boldsymbol{w}^T \overline{\boldsymbol{x}}_n)) + \frac{1}{2} \lambda \boldsymbol{w}^T \boldsymbol{w}$$

Draw a graph with two curves that show how the training accuracy (first curve) is expected to vary with λ and the test accuracy (second curve) is expected to vary with λ.



Question 2 [18 pts] Perceptron. Consider a threshold perceptron with a single unit. Suppose that the weights are initialized to $w_0 = 1, w_1 = 0, w_2 = 0$. Show the weights at each step of the threshold perceptron learning algorithm. A step corresponds to updating the weights based on one data point in the training set $\{(1,1)^T-, (1,2)^T+, (-1,3)^T+\}$. Loop through the training set twice in the following order.

Starting weights: $w_0 = 1,$ $w_1 = 0,$ $w_2 = 0$

-1 -1 -1

Update weights based on $(1,1)^T$-: $w_0 =$ 0 $w_1 =$ -1 $w_2 =$ -1

1 1 2

Update weights based on $(1,2)^T$+: $w_0 =$ 1 $w_1 =$ 0 $w_2 =$ 1

Update weights based on $(-1,3)^T$+: $w_0 =$ 1 $w_1 =$ 0 $w_2 =$ 1

-1 -1 -1

Update weights based on $(1,1)^T$-: $w_0 =$ 0 $w_1 =$ -1 $w_2 =$ 0

1 1 2

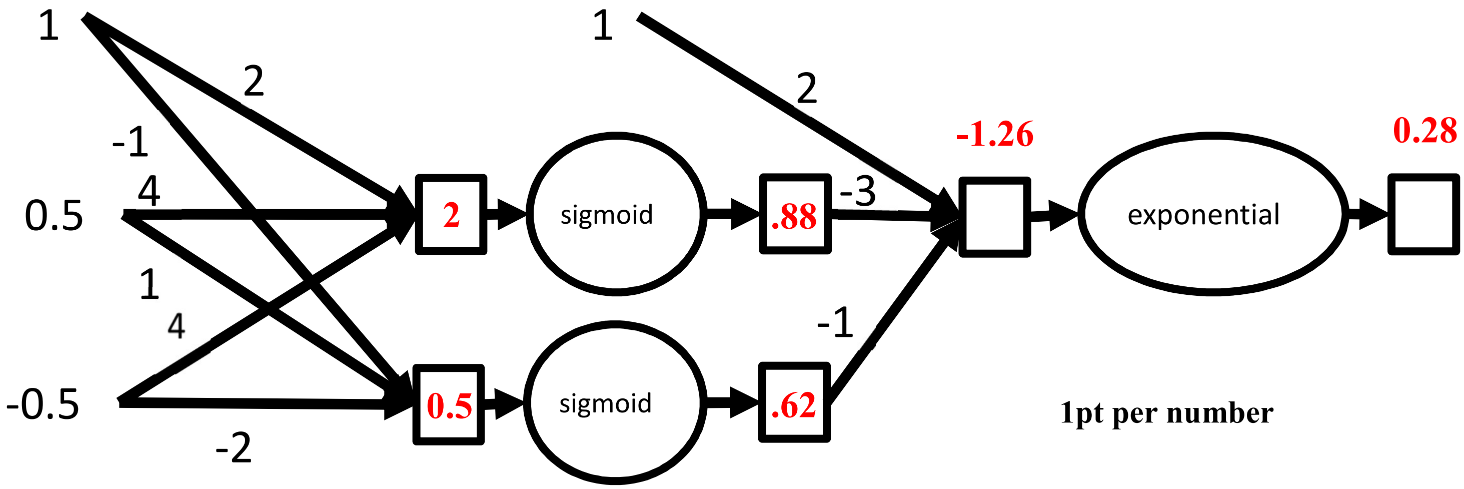
Update weights based on $(1,2)^T$+: $w_0 =$ 1 $w_1 =$ 0 $w_2 =$ 2

Update weights based on $(-1,3)^T$+: $w_0 =$ 1 $w_1 =$ 0 $w_2 =$ 2

1pt per weight

Question 3 [20 pts] Neural networks.

a) **[6 pts]** Forward propagation: Compute the input (a_j) and output (z_j) of each unit j of the following neural network by filling in the empty boxes.



b) **[14 pts]** Backpropagation: Let $\boldsymbol{x}$ be a data point and $f(\boldsymbol{x})$ be the output of the neural network for this data point. Consider the squared loss function $Err(\boldsymbol{x}, y)$ for input $\boldsymbol{x}$ and target y:

$$Err(\boldsymbol{x}, y) = \frac{1}{2}(f(\boldsymbol{x}) - y)^2$$

Indicate how to compute the error δ for each unit in the network:

Output unit k: $\delta_k = h'(a_j)(y_k - z_k) = e^{a_k}(y_k - z_k)$ **(3.5pts)**

Hidden units j: $\delta_j = h'(a_j) \sum_k w_{kj}\delta_k = \sigma(a_j)(1 - \sigma(a_j)) \sum_k w_{kj}\delta_k$ **(3.5pts)**

Indicate how to compute the partial derivatives with respect to each weight w:

Output layer: $\frac{\partial Err}{\partial w_{kj}} = \delta_k z_j = e^{a_k}(y_k - z_k)z_j$ **(3.5pts)**

Hidden layer: $\frac{\partial Err}{\partial w_{ji}} = \delta_j z_i = \sigma(a_j)(1 - \sigma(a_j)) \sum_k w_{kj}\delta_k$ **(3.5pts)**

Question 4 [20 points] Kernels

a) **[8 pts]** Let $\boldsymbol{x} = (x_1, x_2)^T$ and $\boldsymbol{x}' = (x_1', x_2')^T$. Consider the polynomial kernel

$$\begin{aligned} k(\boldsymbol{x}, \boldsymbol{x}') = (\boldsymbol{x}^T \boldsymbol{x} + c)^2 = \emptyset(x)^T \emptyset(x') = (x^T x')^2 + 2x^T x' c + c^2 \\ = (x_1 x_1' + x_2 x_2')^2 + 2(x_1 x_1' + x_2 x_2')c + c^2 \\ = x_1^2 (x_1')^2 + 2x_1 x_2 x_1' x_2' + x_2^2 (x_2')^2 + 2x_1 x_1' c + 2x_2 x_2' c + c^2 \end{aligned}$$

What are the features $\boldsymbol{\phi}(\boldsymbol{x})$ of this kernel?

$$\boldsymbol{\phi}(\boldsymbol{x}) = \begin{pmatrix} x_1^2 \\ \sqrt{2} x_1 x_2 \\ x_2^2 \\ \sqrt{2c} x_1 \\ \sqrt{2c} x_2 \\ c \end{pmatrix}$$

b) **[12 pts]** Let $\phi_1(\boldsymbol{x})$ and $\phi_2(\boldsymbol{x})$ be the features for the kernel

$$k_1(\boldsymbol{x}, \boldsymbol{x}') = \phi_1(\boldsymbol{x})\phi_1(\boldsymbol{x}') + \phi_2(\boldsymbol{x})\phi_2(\boldsymbol{x}')$$

Similarly, let $\phi_3(\boldsymbol{x})$ and $\phi_4(\boldsymbol{x})$ be the features for the kernel

$$k_2(\boldsymbol{x}, \boldsymbol{x}') = \phi_3(\boldsymbol{x})\phi_3(\boldsymbol{x}') + \phi_4(\boldsymbol{x})\phi_4(\boldsymbol{x}')$$

i. What are the features for the kernel $k_3(\boldsymbol{x}, \boldsymbol{x}') = k_1(\boldsymbol{x}, \boldsymbol{x}') + k_2(\boldsymbol{x}, \boldsymbol{x}')$?

$$k_1(\boldsymbol{x}, \boldsymbol{x}') + k_2(\boldsymbol{x}, \boldsymbol{x}') = \phi_1(\boldsymbol{x}') + \phi_2(\boldsymbol{x})\phi_2(\boldsymbol{x}') + \phi_3(\boldsymbol{x})\phi_3(\boldsymbol{x}') + \phi_4(\boldsymbol{x})\phi_4(\boldsymbol{x}')$$

$$\boldsymbol{\phi}(\boldsymbol{x}) = \begin{pmatrix} \phi_1(\boldsymbol{x}) \\ \phi_2(\boldsymbol{x}) \\ \phi_3(\boldsymbol{x}) \\ \phi_4(\boldsymbol{x}) \end{pmatrix} \quad \textbf{(6pts)}$$

ii. What are the features for the kernel $k_4(\boldsymbol{x}, \boldsymbol{x}') = k_1(\boldsymbol{x}, \boldsymbol{x}')k_2(\boldsymbol{x}, \boldsymbol{x}')$?

$$\begin{aligned} k_1(\boldsymbol{x}, \boldsymbol{x}')k_2(\boldsymbol{x}, \boldsymbol{x}') \\ = \phi_1(\boldsymbol{x})\phi_1(\boldsymbol{x}')\phi_3(\boldsymbol{x})\phi_3(\boldsymbol{x}') + \phi_1(\boldsymbol{x})\phi_1(\boldsymbol{x}')\phi_4(\boldsymbol{x})\phi_4(\boldsymbol{x}') \\ + \phi_2(\boldsymbol{x})\phi_2(\boldsymbol{x}')\phi_3(\boldsymbol{x})\phi_3(\boldsymbol{x}') + \phi_2(\boldsymbol{x})\phi_2(\boldsymbol{x}')\phi_4(\boldsymbol{x})\phi_4(\boldsymbol{x}') \end{aligned}$$

$$\boldsymbol{\phi}(\boldsymbol{x}) = \begin{pmatrix} \phi_1(\boldsymbol{x})\phi_3(\boldsymbol{x}) \\ \phi_1(\boldsymbol{x})\phi_4(\boldsymbol{x}) \\ \phi_2(\boldsymbol{x})\phi_3(\boldsymbol{x}) \\ \phi_2(\boldsymbol{x})\phi_4(\boldsymbol{x}) \end{pmatrix} \quad \textbf{(6pts)}$$

Question 5 [20 pts] Indicate whether each statement is true or false. No justification required.

a) **[4 pts]** The computational complexity of generalized linear regression in the feature space is cubic in the number of basis functions while the computational complexity of generalized linear regression in the dual space is cubic in the amount of data.

T

b) **[4 pts]** In binary classification by mixtures of Gaussians with identical covariance matrices, the posterior distribution is a logistic distribution.

T

c) **[4 pts]** Linear regression and logistic regression are special cases of neural networks without any hidden unit.

T

d) **[4 pts]** When measurement noise is Gaussian, linear regression by maximum likelihood yields a different (equal) solution than linear regression by minimum squared loss.

F

e) **[4 pts]** In K-nearest neighbours, using too many neighbours might lead to overfitting (underfitting).

F