Midterm review

For midterm review we'll go through:

1. Important ML concepts
2. Exercises
ML concepts

- **What is supervised learning?**
  Answer: ML setting when our training set consists of inputs and their corresponding labels.

- **Difference between regression / classification?**
  Answer: in *classification* we are predicting a discrete target (like cat or dog class), while in *regression* we are predicting a continuous-valued target (like temperature).

- **What does kNN do?**
  Answer: k Nearest Neighbours is an algorithm that predicts value of a new example based on its k nearest labeled neighbours.
ML concepts

• How does decision tree work?
  Answer: decision trees make predictions by sequentially splitting data on different attributes.

• Name 2 advantages of kNN vs decision tree and vice versa.
  kNN: can incorporate interesting distance measures; few hyperparameters
  decision trees: fast at test time; more interpretable; better deals with missing values.

• What is overfitting and underfitting?
  Overfitting: When the model gets good performance on a particular dataset by "memorizing" it, but fails to generalize to new data.

• Why do we need a validation set?
  Answer: to prevent overfitting.
**ML concepts**

- Based on which measure we can choose a good decision tree split?

**Answer:**

*Fitting* the tree is finding an order to split the data, such that the information gain is maximized at each split.

**Information gain:** tells us how much “information” a feature gives us about the class.

**Entropy:** a measure of impurity, disorder or uncertainty in a set of examples. “How unpredictable a dataset is”.

ML concepts

- Decision boundary of decision trees vs. kNN?
ML concepts

• What does this picture tells us about our data (bias / variance)?

Answer: high bias & low variance.
ML concepts

• Are decision trees and kNN supervised / unsupervised algorithms?

Answer: supervised (we need labels).
ML concepts

• Write a model for binary linear classification ...
  \[ z = \mathbf{w}^T \mathbf{x} + b \]
  \[ y = \begin{cases} 
  1 & \text{if } z \geq r \\
  0 & \text{if } z < r 
\end{cases} \]

• What are the two ways of finding good values for model's parameters \((w, b)\)?
  
  A. direct solution
  
  B. iterative solution (gradient descent)

• What is loss function?

  Answer: it's a function that evaluates how well specific algorithm models the given data; loss function takes predicted values and target values as inputs.
ML concepts

• A loss function for linear classification: 0-1 loss

• Problem?

   Answer: 0-1 loss is bad because it's not informative - its derivative is 0 everywhere it's defined.

\[
L_{0-1}(y, t) = \begin{cases} 
0 & \text{if } y = t \\
1 & \text{if } y \neq t
\end{cases}
\]

Image source:
What are the problems with squared error loss function in classification?

Answer: squared error loss gives a big penalty for correct predictions that are made with high confidence.

A solution?

Predict values only in $[0,1]$ interval. For that we use **sigmoid function** to squash $y$ into $[0,1]$:

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

$$z = \mathbf{w}^\top \mathbf{x} + b$$

$$y = \sigma(z)$$

$$\mathcal{L}_{SE}(y, t) = \frac{1}{2}(y - t)^2.$$
ML concepts

Another solution:
Cross entropy loss.

\[ \sigma(z) = \frac{1}{1 + e^{-z}} \]

\[ z = \mathbf{w}^\top \mathbf{x} + b \]

\[ y = \sigma(z) \]

\[ L_{CE} = -t \log(y) - (1 - t) \log(1 - y) \]
ML concepts

• What is the difference between parameters / hyper parameters of the model?

Answer: **parameters** are learned through training (by iteratively performing gradient descent updates) - weights and biases, **hyperparameters** are "manually" adjusted and set before training - number of hidden layers of a neural network, k for kNN, learning rate etc.

• What is learning rate?

Answer: learning rate is a hyper parameter that controls how much the weights are updated at each iteration. 

\[ w_j \leftarrow w_j - \alpha \frac{\partial J}{\partial w_j} \]

• What if learning rate is too small / too large? (draw a picture)

\[ \alpha \ too\ small: \quad \text{slow progress} \]
\[ \alpha \ too\ large: \quad \text{oscillations} \]
\[ \alpha \ much\ too\ large: \quad \text{instability} \]
ML concepts

• What is regularization? Why do we need it?

Answer: regularization is a technique of adding an extra term to the loss function. It reduces overfitting by keeping the weights of the model smaller.
L1 vs L2 Regularization

L1:

\[ O = \sum_{i=1}^{n} \left( y_i - \sum_{j=1}^{p} X_{i,j} \beta_j \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \]

L2:

\[ O = \sum_{i=1}^{n} \left( y_i - \sum_{j=1}^{p} X_{i,j} \beta_j \right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \]
ML concepts

• What is softmax? Calculate $\text{softmax} \left( \begin{bmatrix} 2 \\ 1 \\ 0.1 \end{bmatrix} \right)$

Answer: softmax is an activation function for multi-class classification that maps input logits to probabilities.

$$\text{softmax} \left( \begin{bmatrix} 2 \\ 1 \\ 0.1 \end{bmatrix} \right) = \begin{bmatrix} \frac{e^2}{e^2 + e^1 + e^{0.1}} \\ \frac{e^1}{e^2 + e^1 + e^{0.1}} \\ \frac{e^{0.1}}{e^2 + e^1 + e^{0.1}} \end{bmatrix} = \begin{bmatrix} 0.7 \\ 0.2 \\ 0.1 \end{bmatrix}$$

$$y_k = \text{softmax}(z_1, \ldots, z_K)_k = \frac{e^{z_k}}{\sum_{k'} e^{z_{k'}}}$$
Other topics to know

• Difference between training, validation and testing sets

• Maximum Likelihood estimation (Slides 3.26-3.30)

• Bagging

• Responsible for material up until slide 5.11
Example 1 - linear classifier weights

Find a linear classifier with weights \( w_1, w_2, w_3, \) and \( b \) which correctly classifies all of these training examples:

<table>
<thead>
<tr>
<th>( x_1 )</th>
<th>( x_2 )</th>
<th>( x_3 )</th>
<th>( t )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

\[ w_1 x_1 + w_2 x_2 + w_3 x_3 + b \geq 0 \]

**Answer:** write a system of inequalities and find one solution (there would be many possible answers).

\[
\begin{align*}
    b & > 0 & b & = 1 \\
    w_2 + b & < 0 & w_1 & = -2 \\
    w_2 + w_3 + b & > 0 & w_2 & = -2 \\
    w_1 + w_2 + w_3 + b & < 0 & w_3 & = 2
\end{align*}
\]
Example 2 - entropy

Suppose binary-valued random variables $X$ and $Y$ have the following joint distribution:

<table>
<thead>
<tr>
<th></th>
<th>$Y = 0$</th>
<th>$Y = 1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$X = 0$</td>
<td>1/8</td>
<td>3/8</td>
</tr>
<tr>
<td>$X = 1$</td>
<td>2/8</td>
<td>2/8</td>
</tr>
</tbody>
</table>

Find **entropy** of a joint distribution $H(X, Y)$ and **conditional entropy** of $Y$ given $X=0$.

**Answer:**

**Entropy of a Joint Distribution**

$$H(X, Y) = - \sum_x \sum_y p(X = x, Y = y) \log_2 p(X = x, Y = y)$$

$$H(X, Y) = - \frac{1}{8} \log_2 \frac{1}{8} - \frac{3}{8} \log_2 \frac{3}{8} - \frac{2}{8} \log_2 \frac{2}{8} - \frac{2}{8} \log_2 \frac{2}{8}$$

**Conditional Entropy**

$$H(Y|X = 0) = - \sum_{y \in \{0,1\}} p(Y = y|X = 0) \cdot \log_2 p(Y = y|X = 0)$$

$$p(Y = 0|X = 0) = \frac{p(Y = 0, X = 0)}{p(X = 0)} = \frac{1/8}{4/8} = \frac{1}{4}$$

$$H(Y|X = 0) = - \frac{1}{4} \log_2 \frac{1}{4} - \frac{3}{4} \log_2 \frac{3}{4}$$
Example 3 - Information Gain

Suppose binary-valued random variables X and Y have the following joint distribution:

<table>
<thead>
<tr>
<th></th>
<th>Y = 0</th>
<th>Y = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>X = 0</td>
<td>1/8</td>
<td>3/8</td>
</tr>
<tr>
<td>X = 1</td>
<td>2/8</td>
<td>2/8</td>
</tr>
</tbody>
</table>

Find information gain \( IG(Y | X) \).

**Answer:** Information Gain: \( IG(Y | X) = H(Y) - H(Y | X) \)

\[
H(Y) = - \sum_y p(Y = y) \log_2 p(Y = y)
\]

\[
H(Y) = - \frac{3}{8} \log_2 \frac{3}{8} - \frac{5}{8} \log_2 \frac{5}{8}
\]

\[
H(Y | X) = p(X = 0) \cdot H(Y | X = 0) + p(X = 1) \cdot H(Y | X = 1)
\]

\[
p(X = 0) = \frac{4}{8} = \frac{1}{2} \quad \quad \quad p(X = 1) = \frac{4}{8} = \frac{1}{2}
\]

\[
H(Y | X = 0) = - \frac{1}{4} \log_2 \frac{1}{4} - \frac{3}{4} \log_2 \frac{3}{4}
\]

\[
H(Y | X = 1) = - \frac{1}{2} \log_2 \frac{1}{2} - \frac{1}{2} \log_2 \frac{1}{2}
\]

plug in values into \( H(Y | X) \) equation