Perceptron:
* If $t_n = +1$, want $w^T \phi_n > 0 \Rightarrow (t_n) w^T \phi_n > 0$, $O = 0$.
* If $t_n = -1$, want $w^T \phi_n < 0 \Rightarrow (t_n) w^T \phi_n < 0 \Rightarrow t_n w^T \phi_n > 0$.
  i.e. we want $t_n w^T \phi_n > 0$ for all samples.
* If this condition is violated for some sample, we want it to be violated by as little as possible, i.e. for misclassified samples, want $t_n w^T \phi_n \leq 0$ as close to 0 as possible, i.e. want to maximize $t_n w^T \phi_n$ or minimize $-t_n w^T \phi_n$.

\[ \frac{\partial}{\partial w} \left[ -t_n w^T \phi_n \right] = -t_n \phi_n \]

Bayes theorem:
\[ P(A|B) = \frac{P(B|A)P(A)}{P(B)} = \frac{P(B|A)P(A)}{\sum_i P(B|A_i)P(A_i)} \]

Maximum likelihood estimation:
\[ w^* = \arg \max_w P \left( \text{Data} \mid w \right) \]

* Example: We want a model coin tosses. The underlying model $w = P(H|W)$.
  Then $w^* = \arg \max_w L(w)$ where $L(w) = P(S, H, T, T, H, T, H, T, T | w) = P(H|W)^N H P(T|W)^N T$.

\[ \frac{\partial}{\partial w} \left[ \log L(w) \right] = 0 = \frac{\partial}{\partial w} \left[ N_H \log P(H|W) + N_T \log P(T|W) \right] = \frac{\partial}{\partial w} \left[ N_H \log w + N_T \log (1-w) \right] \\
= \frac{N_H}{w} - \frac{N_T}{1-w} \Rightarrow N_H - N_H w - N_T w = 0 \Rightarrow w = \frac{N_H}{N_H + N_T} \quad \text{(as expected!)} \]
Logistic regression

* \( P(y_i = 1 | x_i) = \frac{1}{1 + e^{-w^T x_i}} = \sigma(w^T x_i) \) where \( \sigma \) is the logistic sigmoid.

* Decision boundary: \( P(y = 1|x) > P(y=0|x) \Rightarrow \frac{P(y=1|x)}{P(y=0|x)} > 1 \)
  \[ \Rightarrow \log \left[ \frac{P(y=1|x)}{P(y=0|x)} \right] > 0 \Rightarrow \log \left[ \frac{\frac{P(y=1|x)}{1-P(y=1|x)}}{1} \right] > 0 \]
  \[ \Rightarrow \log \left[ \frac{1}{1+e^{-w^T x}} \right] > 0 \Rightarrow \log \left[ \frac{1}{1+e^{-w^T x}} \frac{1}{1+e^{-w^T x}} \right] > 0 \Rightarrow \]
  \[ \Rightarrow \log \left[ \frac{1}{e^{-w^T x}} \right] > 0 \Rightarrow -\log e^{-w^T x} > 0 \Rightarrow w^T x > 0 \] (linear classifier)

* Solution for \( w: \) \( w^* = \arg \max_w L(w) \) where

\[
L(w) = \prod_{i=1}^N P(y_i | w, x_i) = \prod_{i=1}^N \sigma(w^T x_i)^{y_i} (1-\sigma(w^T x_i))^{(1-y_i)}
\]

\[
\log L(w) = \sum_{i=1}^N y_i \log \sigma(w^T x_i) + (1-y_i) \log (1-\sigma(w^T x_i))
\]

\[
\frac{dL(w)}{dw} = 0 = \sum_{i=1}^N y_i \frac{1}{\sigma(w^T x_i)} \frac{d\sigma(w^T x_i)}{dw} + (1-y_i) \frac{1}{1-\sigma(w^T x_i)} \frac{d\sigma(w^T x_i)}{dw}
\]

\[
= \sum_{i=1}^N \left( y_i \frac{\sigma(w^T x_i) - \sigma(w^T x_i)}{\sigma(w^T x_i)(1-\sigma(w^T x_i))} + y_i \frac{\sigma(w^T x_i) - \sigma(w^T x_i)}{\sigma(w^T x_i)(1-\sigma(w^T x_i))} \right) \frac{d\sigma(w^T x_i)}{dw}
\]

\[
= \sum_{i=1}^N \left( y_i - \sigma(w^T x_i) \right) x_i
\]

Prediction error = 0 if \( y_i = 1, P(y_i = 1 | x_i) = 1 \) or \( y_i = 0, P(y_i = 1 | x_i) = 0 \)

* Gradient ascent update: \( w^{(t+1)} = w^{(t)} + \gamma \frac{dL(w)}{dw} \)

\[
w^{(t+1)} = w^{(t)} + \gamma \sum_{i=1}^N \left( y_i - \sigma(w^{(t)^T} x_i) \right) x_i
\]