Logistic Regression

Machine Learning

THE UNIVERSITY OF UTAH

Where are we?

We have seen the following ideas

- Linear models
- Learning as loss minimization
- Bayesian learning criteria (MAP and MLE estimation)
- The Naïve Bayes classifier

This lecture

- Logistic regression
- Connection to Naïve Bayes
- Training a logistic regression classifier
- Back to loss minimization

This lecture

- Logistic regression
- Connection to Naïve Bayes
- Training a logistic regression classifier
- Back to loss minimization

Logistic Regression: Setup

- The setting
  - Binary classification
  - Inputs: Feature vectors $\mathbf{x} \in \Re^{\mathrm{d}}$
  - Labels: $\mathbf{y} \in \{-1, +1\}$

- Training data
  - S = $\{(\mathbf{x}_i, \mathbf{y}_i)\}$, m examples

Classification, but…

The output $\mathbf{y}$ is discrete valued (-1 or 1)

Instead of predicting the output, let us try to predict P(y = 1 | $\mathbf{x}$)

Classification, but…

The output $\mathbf{y}$ is discrete valued (-1 or 1)

Instead of predicting the output, let us try to predict P(y = 1 | **x**)

Expand hypothesis space to functions whose output is [0-1]

- Original problem: $\Re^{d} \to \{-1, 1\}$
- Modified problem: $\Re^{d} \to [0\text{-}1]$
- Effectively make the problem a regression problem

Many hypothesis spaces possible

The Sigmoid function

The hypothesis space for logistic regression: All functions of the form

$$h_{\mathbf{w}}(\mathbf{x}) = \sigma(\mathbf{w}^T\mathbf{x}) = \frac{1}{1 + \exp(-\mathbf{w}^T\mathbf{x})}$$

The Sigmoid function

The hypothesis space for logistic regression: All functions of the form

$$h_{\mathbf{w}}(\mathbf{x}) = \sigma(\mathbf{w}^T\mathbf{x}) = \frac{1}{1 + \exp(-\mathbf{w}^T\mathbf{x})}$$

That is, a linear function, composed with a sigmoid function (the logistic function) σ

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

This is a reasonable choice. We will see why later

The Sigmoid function

The hypothesis space for logistic regression: All functions of the form

$$h_{\mathbf{w}}(\mathbf{x}) = \sigma(\mathbf{w}^T\mathbf{x}) = \frac{1}{1 + \exp(-\mathbf{w}^T\mathbf{x})}$$

That is, a linear function, composed with a sigmoid function (the logistic function) σ

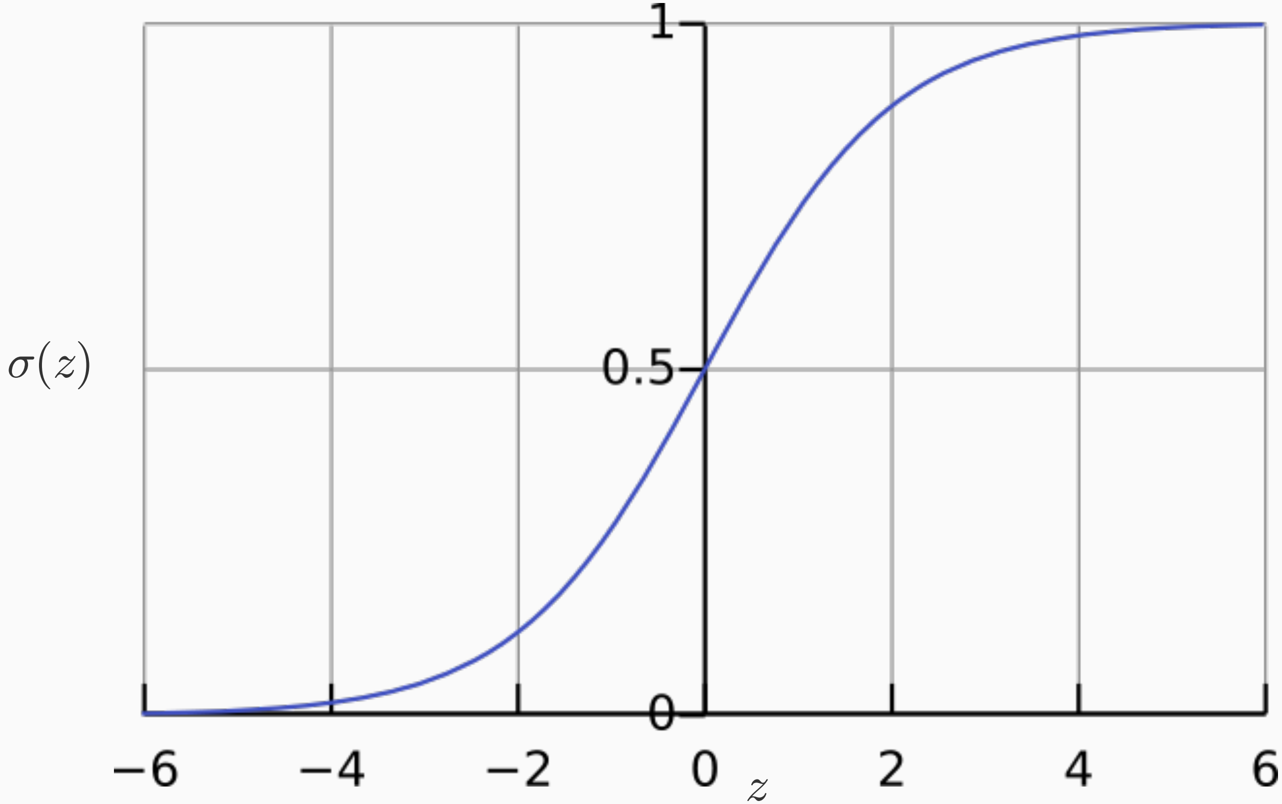
$$\sigma(z) = \frac{1}{1 + \exp(-z)}$$

What is the domain and the range of the sigmoid function?

This is a reasonable choice. We will see why later

The Sigmoid function

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



The Sigmoid function

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

What is its derivative with respect to z?

$$\frac{d\sigma}{dz} \quad = \quad \frac{d}{dz} \frac{1}{1 + \exp(-z)}$$

The Sigmoid function

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

What is its derivative with respect to z?

$$\begin{aligned}
\frac{d\sigma}{dz} &= \frac{d}{dz}\frac{1}{1+\exp(-z)} \\
&= \frac{1}{(1+\exp(-z))^2} \cdot \exp(-z) \\
&= \left(1 - \frac{1}{1+\exp(-z)}\right) \cdot \frac{1}{1+\exp(-z)} \\
&= \sigma(z)\left(1 - \sigma(z)\right).
\end{aligned}$$

Predicting probabilities

According to the logistic regression model, we have

$$P(y = 1|\mathbf{x}; \mathbf{w}) = \sigma(\mathbf{w}^T\mathbf{x}) = \frac{1}{1 + \exp(-\mathbf{w}^T\mathbf{x})}$$

$$P(y = -1|\mathbf{x}; \mathbf{w}) = 1 - \sigma(\mathbf{w}^T\mathbf{x}) = \frac{\exp(-\mathbf{w}^T\mathbf{x})}{1 + \exp(-\mathbf{w}^T\mathbf{x})}$$

Predicting probabilities

According to the logistic regression model, we have

$$P(y = 1|\mathbf{x}; \mathbf{w}) = \sigma(\mathbf{w}^T\mathbf{x}) = \frac{1}{1 + \exp(-\mathbf{w}^T\mathbf{x})}$$

$$P(y = -1|\mathbf{x}; \mathbf{w}) = 1 - \sigma(\mathbf{w}^T\mathbf{x}) = \boxed{\frac{\exp(-\mathbf{w}^T\mathbf{x})}{1 + \exp(-\mathbf{w}^T\mathbf{x})}}$$

$$\frac{1}{1 + \exp(\mathbf{w}^T\mathbf{x})}$$

Predicting probabilities

According to the logistic regression model, we have

$$P(y = 1|\mathbf{x}; \mathbf{w}) = \sigma(\mathbf{w}^T \mathbf{x}) = \frac{1}{1 + \exp(-\mathbf{w}^T \mathbf{x})}$$

$$P(y = -1|\mathbf{x}; \mathbf{w}) = 1 - \sigma(\mathbf{w}^T \mathbf{x}) = \frac{1}{1 + \exp(\mathbf{w}^T \mathbf{x})}$$

Predicting probabilities

According to the logistic regression model, we have

$$P(y = 1|\mathbf{x}; \mathbf{w}) = \sigma(\mathbf{w}^T\mathbf{x}) = \frac{1}{1 + \exp(-\mathbf{w}^T\mathbf{x})}$$

$$P(y = -1|\mathbf{x}; \mathbf{w}) = 1 - \sigma(\mathbf{w}^T\mathbf{x}) = \frac{1}{1 + \exp(\mathbf{w}^T\mathbf{x})}$$

Or equivalently

$$P(y|\mathbf{x}; \mathbf{w}) = \frac{1}{1 + \exp(-y\mathbf{w}^T\mathbf{x})}$$

Predicting probabilities

According to the logistic regression model, we have

$$P(y = 1|\mathbf{x}; \mathbf{w}) = \sigma(\mathbf{w}^T\mathbf{x}) = \frac{1}{1 + \exp(-\mathbf{w}^T\mathbf{x})}$$

$$P(y = -1|\mathbf{x}; \mathbf{w}) = 1 - \sigma(\mathbf{w}^T\mathbf{x}) = \frac{1}{1 + \exp(\mathbf{w}^T\mathbf{x})}$$

Note that we are directly modeling $P(y \mid x)$ rather than $P(x \mid y)$ and $P(y)$

Or equivalently

$$P(y|\mathbf{x}; \mathbf{w}) = \frac{1}{1 + \exp(-y\mathbf{w}^T\mathbf{x})}$$

Predicting a label with logistic regression

$$P(y = 1|\mathbf{x}; \mathbf{w}) = \sigma(\mathbf{w}^T \mathbf{x}) = \frac{1}{1 + \exp(-\mathbf{w}^T \mathbf{x})}$$

- Compute P(y =1 | x; w)

- If this is greater than half, predict 1 else predict -1
  - What does this correspond to in terms of $\mathbf{w}^T\mathbf{x}$?

Predicting a label with logistic regression

$$P(y = 1|\mathbf{x};\mathbf{w}) = \sigma(\mathbf{w}^T\mathbf{x}) = \frac{1}{1 + \exp(-\mathbf{w}^T\mathbf{x})}$$

- Compute P(y =1 | x; w)

- If this is greater than half, predict 1 else predict -1
  - What does this correspond to in terms of $\mathbf{w}^T\mathbf{x}$?

  - Prediction = sgn($\mathbf{w}^T\mathbf{x}$)

This lecture

- Logistic regression
- Connection to Naïve Bayes
- Training a logistic regression classifier
- Back to loss minimization

Naïve Bayes and Logistic regression

Remember that the naïve Bayes decision is a linear function

$$\log \frac{P(y=-1|\mathbf{x},\mathbf{w})}{P(y=+1|\mathbf{x},\mathbf{w})} = \mathbf{w}^T\mathbf{x}$$

Here, the P's represent the Naïve Bayes posterior distribution, and **w** can be used to calculate the priors and the likelihoods.

That is, $P(y = 1 \mid \mathbf{w}, \mathbf{x})$ is computed using $P(\mathbf{x} \mid y = 1, \mathbf{w})$ and $P(y = 1 \mid \mathbf{w})$

Naïve Bayes and Logistic regression

Remember that the naïve Bayes decision is a linear function

$$\log \frac{P(y=-1|\mathbf{x},\mathbf{w})}{P(y=+1|\mathbf{x},\mathbf{w})} = \mathbf{w}^T\mathbf{x}$$

But we also know that $P(y=+1|\mathbf{x},\mathbf{w}) = 1 - P(y=-1|\mathbf{x},\mathbf{w})$

Naïve Bayes and Logistic regression

Remember that the naïve Bayes decision is a linear function

$$\log \frac{P(y=-1|\mathbf{x},\mathbf{w})}{P(y=+1|\mathbf{x},\mathbf{w})} = \mathbf{w}^T\mathbf{x}$$

But we also know that $P(y=+1|\mathbf{x},\mathbf{w}) = 1 - P(y=-1|\mathbf{x},\mathbf{w})$

Substituting in the above expression, we get

$$P(y=+1|\mathbf{w},\mathbf{x}) = \sigma(\mathbf{w}^T\mathbf{x}) = \frac{1}{1+\exp(-\mathbf{w}^T\mathbf{x})}$$

Naïve Bayes and Logistic regression

Remember that the naïve Bayes decision is a linear function

$$\log \frac{P(y = -1|\mathbf{x}, \mathbf{w})}{P(y = +1|\mathbf{x}, \mathbf{w})} = \mathbf{w}^T\mathbf{x}$$

That is, both naïve Bayes and logistic regression try to compute the same *posterior distribution* over the outputs

But we ... $|\mathbf{x}, \mathbf{w})$

Naïve Bayes is a generative model.

Substit...

Logistic Regression is the discriminative version.

$$P(y = +1|\mathbf{w}, \mathbf{x}) = \sigma(\mathbf{w}^T\mathbf{x}) = \frac{1}{1 + \exp(-\mathbf{w}^T\mathbf{x})}$$

This lecture

- Logistic regression
- Connection to Naïve Bayes
- Training a logistic regression classifier
  - First: Maximum likelihood estimation
  - Then: Adding priors → Maximum a Posteriori estimation
- Back to loss minimization

Maximum likelihood estimation

Let's get back to the problem of learning

- Training data
  - S = $\{(\mathbf{x}_i, \mathbf{y}_i)\}$, m examples
- What we want
  - Find a **w** such that P(S | **w**) is maximized
  - We know that our examples are drawn independently and are identically distributed (i.i.d)
  - How do we proceed?

Maximum likelihood estimation

$$\operatorname*{argmax}_{\mathbf{w}} P(S|\mathbf{w}) = \operatorname*{argmax}_{\mathbf{w}} \prod_{i=1}^{m} P(y_i|\mathbf{x}_i, \mathbf{w})$$

The usual trick: Convert products to sums by taking log

Recall that this works only because log is an increasing function and the maximizer will not change

Maximum likelihood estimation

$$\operatorname*{argmax}_{\mathbf{w}} P(S|\mathbf{w}) = \operatorname*{argmax}_{\mathbf{w}} \prod_{i=1}^{m} P(y_i|\mathbf{x}_i, \mathbf{w})$$

$$\Big\downarrow \boxed{\text{Equivalent to solving}}$$

$$\max_{\mathbf{w}} \sum_{i}^{m} \log P(y_i|\mathbf{x}_i, \mathbf{w})$$

Maximum likelihood estimation

$$\underset{\mathbf{w}}{\operatorname{argmax}} P(S|\mathbf{w}) = \underset{\mathbf{w}}{\operatorname{argmax}} \prod_{i=1}^{m} P(y_i|\mathbf{x}_i, \mathbf{w})$$

$$\max_{\mathbf{w}} \sum_{i}^{m} \log P(y_i|\mathbf{x}_i, \mathbf{w})$$

But (by definition) we know that

$$P(y|\mathbf{w}, \mathbf{x}) = \sigma(y_i \mathbf{w}^T \mathbf{x}_i) = \frac{1}{1 + \exp(-y_i \mathbf{w}^T \mathbf{x}_i)}$$

$$P(y|\mathbf{w},\mathbf{x}) = \frac{1}{1+\exp(-y_i\mathbf{w}^T\mathbf{x}_i)}$$

Maximum likelihood estimation

$$\operatorname*{argmax}_{\mathbf{w}} P(S|\mathbf{w}) = \operatorname*{argmax}_{\mathbf{w}} \prod_{i=1}^{m} P(y_i|\mathbf{x}_i,\mathbf{w})$$

$$\max_{\mathbf{w}} \sum_{i}^{m} \log P(y_i|\mathbf{x}_i,\mathbf{w})$$

Equivalent to solving

$$\max_{\mathbf{w}} \sum_{i}^{m} -\log(1+\exp(-y_i\mathbf{w}^T\mathbf{x}_i))$$

$$P(y|\mathbf{w},\mathbf{x}) = \frac{1}{1+\exp(-y_i\mathbf{w}^T\mathbf{x}_i)}$$

Maximum likelihood estimation

$$\underset{\mathbf{w}}{\operatorname{argmax}}\, P(S|\mathbf{w}) = \underset{\mathbf{w}}{\operatorname{argmax}} \prod_{i=1}^{m} P(y_i|\mathbf{x}_i,\mathbf{w})$$

The goal: Maximum likelihood training of a discriminative probabilistic classifier under the logistic model for the posterior distribution.

$$\max_{\mathbf{w}} \sum_{i}^{m} \log P(y_i|\mathbf{x}_i,\mathbf{w})$$

Equivalent to solving

$$\max_{\mathbf{w}} \sum_{i}^{m} -\log(1+\exp(-y_i\mathbf{w}^T\mathbf{x}_i))$$

$$P(y|\mathbf{w},\mathbf{x}) = \frac{1}{1+\exp(-\mathrm{y_i}\mathbf{w}^T\mathbf{x}_i)}$$

Maximum likelihood estimation

$$\operatorname*{argmax}_{\mathbf{w}} P(S|\mathbf{w}) = \operatorname*{argmax}_{\mathbf{w}} \prod_{i=1}^{m} P(y_i|\mathbf{x}_i,\mathbf{w})$$

The goal: Maximum likelihood training of a discriminative probabilistic classifier under the logistic model for the posterior distribution.

$$\max_{\mathbf{w}} \sum_{i}^{m} \log P(y_i|\mathbf{x}_i,\mathbf{w})$$

Equivalent to solving

$$\max_{\mathbf{w}} \sum_{i}^{m} -\log(1+\exp(-y_i\mathbf{w}^T\mathbf{x}_i))$$

Equivalent to: Training a linear classifier by minimizing the ***logistic loss***.

Maximum a posteriori estimation

We could also add a prior on the weights

Suppose each weight in the weight vector is drawn independently from the normal distribution with zero mean and standard deviation σ

$$p(\mathbf{w}) = \prod_{j=1}^{d} p(w_i) = \prod_{j=1}^{d} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-w_i^2}{\sigma^2}\right)$$

MAP estimation for logistic regression

Maximum likelihood estimation

$$\arg\max_{\mathbf{w}} P(S|\mathbf{w}) = \arg\max_{\mathbf{w}} \prod_{i=1}^{m} P(y_i|\mathbf{x}_i, \mathbf{w})$$

$$\max_{\mathbf{w}} \sum_{i=1}^{m} \log P(y_i|\mathbf{x}_i, \mathbf{w})$$

Equivalent to solving

$$\max_{\mathbf{w}} \sum_{i=1}^{m} -\log\left(1 + \exp(-y_i \mathbf{w}^T \mathbf{x}_i)\right)$$

$$p(\mathbf{w}) = \prod_{j=1}^{d} p(w_i) = \prod_{j=1}^{d} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-w_i^2}{\sigma^2}\right)$$

Let us work through this procedure again to see what changes

MAP estimation for logistic regression

Maximum likelihood estimation

$$\arg\max_{\mathbf{w}} P(S|\mathbf{w}) = \arg\max_{\mathbf{w}} \prod_{i=1}^{m} P(y_i|\mathbf{x}_i, \mathbf{w})$$

$$\max_{\mathbf{w}} \sum_{i=1}^{m} \log P(y_i|\mathbf{x}_i, \mathbf{w})$$

Equivalent to solving

$$\max_{\mathbf{w}} \sum_{i=1}^{m} -\log\left(1 + \exp(-y_i \mathbf{w}^T \mathbf{x}_i)\right)$$

$$p(\mathbf{w}) = \prod_{j=1}^{d} p(w_i) = \prod_{j=1}^{d} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-w_i^2}{\sigma^2}\right)$$

Let us work through this procedure again to see what changes

What is the goal of MAP estimation? (In maximum likelihood, we maximized the likelihood of the data)

MAP estimation for logistic regression

Maximum likelihood estimation

$$\arg\max_{\mathbf{w}} P(S|\mathbf{w}) = \arg\max_{\mathbf{w}} \prod_{i=1}^{m} P(y_i|\mathbf{x}_i, \mathbf{w})$$

$$\max_{\mathbf{w}} \sum_{i=1}^{m} \log P(y_i|\mathbf{x}_i, \mathbf{w})$$

Equivalent to solving

$$\max_{\mathbf{w}} \sum_{i=1}^{m} -\log\left(1 + \exp(-y_i \mathbf{w}^T \mathbf{x}_i)\right)$$

$$p(\mathbf{w}) = \prod_{j=1}^{d} p(w_i) = \prod_{j=1}^{d} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-w_i^2}{\sigma^2}\right)$$

What is the goal of MAP estimation? (In maximum likelihood, we maximized the likelihood of the data)

To maximize the posterior probability of the model given the data (i.e. to find the most probable model, given the data)

$$P(\mathbf{w}|S) \propto P(S|\mathbf{w})P(\mathbf{w})$$

MAP estimation for logistic regression

Maximum likelihood estimation

$$\arg\max_{\mathbf{w}} P(S|\mathbf{w}) = \arg\max_{\mathbf{w}} \prod_{i=1}^{m} P(y_i|\mathbf{x}_i, \mathbf{w})$$

$$\max_{\mathbf{w}} \sum_{i=1}^{m} \log P(y_i|\mathbf{x}_i, \mathbf{w})$$

Equivalent to solving

$$\max_{\mathbf{w}} \sum_{i=1}^{m} -\log\left(1 + \exp(-y_i \mathbf{w}^T \mathbf{x}_i)\right)$$

$$p(\mathbf{w}) = \prod_{j=1}^{d} p(w_i) = \prod_{j=1}^{d} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-w_i^2}{\sigma^2}\right)$$

Learning by solving

$$\operatorname*{argmax}_{\mathbf{w}} P(\mathbf{w}|S) = \operatorname*{argmax}_{\mathbf{w}} P(S|\mathbf{w})P(\mathbf{w})$$

MAP estimation for logistic regression

Maximum likelihood estimation

$$\arg\max_{\mathbf{w}} P(S|\mathbf{w}) = \arg\max_{\mathbf{w}} \prod_{i=1}^{m} P(y_i|\mathbf{x}_i, \mathbf{w})$$

$$\max_{\mathbf{w}} \sum_{i=1}^{m} \log P(y_i|\mathbf{x}_i, \mathbf{w})$$

$\boxed{\text{Equivalent to solving}}$

$$\max_{\mathbf{w}} \sum_{i=1}^{m} -\log\left(1 + \exp(-y_i \mathbf{w}^T \mathbf{x}_i)\right)$$

$$p(\mathbf{w}) = \prod_{j=1}^{d} p(w_i) = \prod_{j=1}^{d} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-w_i^2}{\sigma^2}\right)$$

Learning by solving

$$\operatorname*{argmax}_{\mathbf{w}} P(S|\mathbf{w})P(\mathbf{w})$$

Take log to simplify

$$\max_{\mathbf{w}} \log P(S|\mathbf{w}) + \log P(\mathbf{w})$$

MAP estimation for logistic regression

Maximum likelihood estimation

$$\arg\max_{\mathbf{w}} P(S|\mathbf{w}) = \arg\max_{\mathbf{w}} \prod_{i=1}^{m} P(y_i|\mathbf{x}_i, \mathbf{w})$$

$$\max_{\mathbf{w}} \sum_{i=1}^{m} \log P(y_i|\mathbf{x}_i, \mathbf{w})$$

Equivalent to solving

$$\max_{\mathbf{w}} \sum_{i=1}^{m} -\log\left(1 + \exp(-y_i \mathbf{w}^T \mathbf{x}_i)\right)$$

$$p(\mathbf{w}) = \prod_{j=1}^{d} p(w_i) = \prod_{j=1}^{d} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-w_i^2}{\sigma^2}\right)$$

Learning by solving

$$\operatorname*{argmax}_{\mathbf{w}} P(S|\mathbf{w})P(\mathbf{w})$$

Take log to simplify

$$\max_{\mathbf{w}} \log P(S|\mathbf{w}) + \log P(\mathbf{w})$$

We have already expanded out the first term.

$$\sum_{i}^{m} -\log(1 + \exp(-y_i \mathbf{w}^T \mathbf{x}_i))$$

MAP estimation for logistic regression

Maximum likelihood estimation

$$\arg\max_{\mathbf{w}} P(S|\mathbf{w}) = \arg\max_{\mathbf{w}} \prod_{i=1}^{m} P(y_i|\mathbf{x}_i, \mathbf{w})$$

$$\max_{\mathbf{w}} \sum_{i=1}^{m} \log P(y_i|\mathbf{x}_i, \mathbf{w})$$

Equivalent to solving

$$\max_{\mathbf{w}} \sum_{i=1}^{m} -\log\left(1 + \exp(-y_i \mathbf{w}^T \mathbf{x}_i)\right)$$

$$p(\mathbf{w}) = \prod_{j=1}^{d} p(w_i) = \prod_{j=1}^{d} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-w_i^2}{\sigma^2}\right)$$

Learning by solving

$$\underset{\mathbf{w}}{\operatorname{argmax}}\, P(S|\mathbf{w}) P(\mathbf{w})$$

Take log to simplify

$$\max_{\mathbf{w}} \log P(S|\mathbf{w}) + \log P(\mathbf{w})$$

Expand the log prior

$$\sum_{i}^{m} -\log(1 + \exp(-y_i \mathbf{w}^T \mathbf{x}_i)) + \sum_{j=1}^{d} \frac{-w_i^2}{\sigma^2} + constants$$

MAP estimation for logistic regression

Maximum likelihood estimation

$$\arg\max_{\mathbf{w}} P(S|\mathbf{w}) = \arg\max_{\mathbf{w}} \prod_{i=1}^{m} P(y_i|\mathbf{x}_i, \mathbf{w})$$

$$\max_{\mathbf{w}} \sum_{i=1}^{m} \log P(y_i|\mathbf{x}_i, \mathbf{w})$$

Equivalent to solving

$$\max_{\mathbf{w}} \sum_{i=1}^{m} -\log\left(1 + \exp(-y_i \mathbf{w}^T \mathbf{x}_i)\right)$$

$$p(\mathbf{w}) = \prod_{j=1}^{d} p(w_i) = \prod_{j=1}^{d} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-w_i^2}{\sigma^2}\right)$$

Learning by solving

$$\operatorname*{argmax}_{\mathbf{w}} P(S|\mathbf{w}) P(\mathbf{w})$$

Take log to simplify

$$\max_{\mathbf{w}} \log P(S|\mathbf{w}) + \log P(\mathbf{w})$$

$$\max_{\mathbf{w}} \sum_{i}^{m} -\log(1 + \exp(-y_i \mathbf{w}^T \mathbf{x}_i) + \sum_{j=1}^{d} \frac{-w_i^2}{\sigma^2} + constants$$

MAP estimation for logistic regression

Maximum likelihood estimation

$$\arg\max_{\mathbf{w}} P(S|\mathbf{w}) = \arg\max_{\mathbf{w}} \prod_{i=1}^{m} P(y_i|\mathbf{x}_i, \mathbf{w})$$

$$\max_{\mathbf{w}} \sum_{i=1}^{m} \log P(y_i|\mathbf{x}_i, \mathbf{w})$$

Equivalent to solving

$$\max_{\mathbf{w}} \sum_{i=1}^{m} -\log\left(1 + \exp(-y_i \mathbf{w}^T \mathbf{x}_i)\right)$$

$$p(\mathbf{w}) = \prod_{j=1}^{d} p(w_i) = \prod_{j=1}^{d} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-w_i^2}{\sigma^2}\right)$$

Learning by solving

$$\underset{\mathbf{w}}{\text{argmax}}\, P(S|\mathbf{w}) P(\mathbf{w})$$

Take log to simplify

$$\max_{\mathbf{w}} \log P(S|\mathbf{w}) + \log P(\mathbf{w})$$

$$\max_{\mathbf{w}} \sum_{i}^{m} -\log(1 + \exp(-y_i \mathbf{w}^T \mathbf{x}_i) - \frac{1}{\sigma^2} \mathbf{w}^T \mathbf{w}$$

MAP estimation for logistic regression

Maximum likelihood estimation

$$\arg\max_{\mathbf{w}} P(S|\mathbf{w}) = \arg\max_{\mathbf{w}} \prod_{i=1}^{m} P(y_i|\mathbf{x}_i, \mathbf{w})$$

$$\max_{\mathbf{w}} \sum_{i=1}^{m} \log P(y_i|\mathbf{x}_i, \mathbf{w})$$

Equivalent to solving

$$\max_{\mathbf{w}} \sum_{i=1}^{m} - \log\left(1 + \exp(-y_i \mathbf{w}^T \mathbf{x}_i)\right)$$

$$p(\mathbf{w}) = \prod_{j=1}^{d} p(w_i) = \prod_{j=1}^{d} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-w_i^2}{\sigma^2}\right)$$

Learning by solving

$$\underset{\mathbf{w}}{\operatorname{argmax}}\, P(S|\mathbf{w})P(\mathbf{w})$$

Take log to simplify

$$\max_{\mathbf{w}} \log P(S|\mathbf{w}) + \log P(\mathbf{w})$$

$$\max_{\mathbf{w}} \sum_{i}^{m} -\log(1 + \exp(-y_i \mathbf{w}^T \mathbf{x}_i) - \frac{1}{\sigma^2} \mathbf{w}^T \mathbf{w}$$

Maximizing a negative function is the same as minimizing the function

Learning a logistic regression classifier

Learning a logistic regression classifier is equivalent to solving

$$\min_{\mathbf{w}} \sum_{i}^{m} \log(1 + \exp(-y_i \mathbf{w}^T \mathbf{x}_i) + \frac{1}{\sigma^2} \mathbf{w}^T \mathbf{w}$$

Learning a logistic regression classifier

Learning a logistic regression classifier is equivalent to solving

$$\min_{\mathbf{w}} \sum_{i}^{m} \log(1 + \exp(-y_i \mathbf{w}^T \mathbf{x}_i) + \frac{1}{\sigma^2} \mathbf{w}^T \mathbf{w}$$

Where have we seen this before?

Learning a logistic regression classifier

Learning a logistic regression classifier is equivalent to solving

$$\min_{\mathbf{w}} \sum_{i}^{m} \log(1 + \exp(-y_i \mathbf{w}^T \mathbf{x}_i) + \frac{1}{\sigma^2} \mathbf{w}^T \mathbf{w}$$

Where have we seen this before?

The first question in the homework: Write down the stochastic gradient descent algorithm for this?

Historically, other training algorithms exist. In particular, you might run into LBFGS

Logistic regression is…

- A classifier that predicts the probability that the label is +1 for a particular input

- The discriminative counter-part of the naïve Bayes classifier

- A discriminative classifier that can be trained via MAP or MLE estimation

- A discriminative classifier that minimizes the logistic loss over the training set

This lecture

- Logistic regression
- Connection to Naïve Bayes
- Training a logistic regression classifier
- Back to loss minimization

Learning as loss minimization

- The setup
  - Examples $\mathbf{x}$ drawn from a fixed, unknown distribution D
  - Hidden oracle classifier f labels examples
  - We wish to find a hypothesis h that mimics f
- The ideal situation
  - Define a function L that penalizes bad hypotheses
  - **Learning:** Pick a function h $\in$ H to minimize expected loss

$$\min_{h \in H} E_{\mathbf{x} \sim D} \left[L\left(h(\mathbf{x}), f(\mathbf{x}) \right) \right]$$

But distribution D is unknown

- Instead, minimize *empirical loss* on the training set

$$\min_{h \in H} \frac{1}{m} \sum_i L(h(\mathbf{x}_i), f(\mathbf{x}_i))$$

Empirical loss minimization

Learning = minimize *empirical loss* on the training set

$$\min_{h \in H} \frac{1}{m} \sum_i L(h(\mathbf{x}_i), f(\mathbf{x}_i))$$

Is there a problem here?

Empirical loss minimization

Learning = minimize *empirical loss* on the training set

$$\min_{h \in H} \frac{1}{m} \sum_i L(h(\mathbf{x}_i), f(\mathbf{x}_i))$$

Is there a problem here? Overfitting!

We need something that biases the learner towards simpler hypotheses

- Achieved using a regularizer, which penalizes complex hypotheses

Regularized loss minimization

- Learning: $\min_{h \in H} \text{regularizer}(h) + C\frac{1}{m}\sum_i L(h(\mathbf{x}_i), f(\mathbf{x}_i))$
- With linear classifiers: $\min_{\mathbf{w}} \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_i L(y_i, \mathbf{x}_i, \mathbf{w})$
  (using $l2$ regularization)
- What is a loss function?
  - Loss functions should penalize mistakes
  - We are minimizing average loss over the training data
- What is the ideal loss function for classification?

The 0-1 loss

Penalize classification mistakes between true label y and prediction y'

$$L_{0\text{-}1}(y, y') = \begin{cases} 1 & \text{if } y \neq y', \\ 0 & \text{if } y = y'. \end{cases}$$

- For linear classifiers, the prediction y' = sgn($\mathbf{w}^T\mathbf{x}$)
  - Mistake if y $\mathbf{w}^T\mathbf{x} \leq 0$

$$L_{0\text{-}1}(y, \mathbf{x}, \mathbf{w}) = \begin{cases} 1 & \text{if } y\ \mathbf{w}^T\mathbf{x} \leq 0, \\ 0 & \text{otherwise.} \end{cases}$$

Minimizing 0-1 loss is intractable. Need surrogates

$$\min_{h \in H} \text{regularizer}(h) + C\frac{1}{m}\sum_i L(h(\mathbf{x}_i), f(\mathbf{x}_i))$$

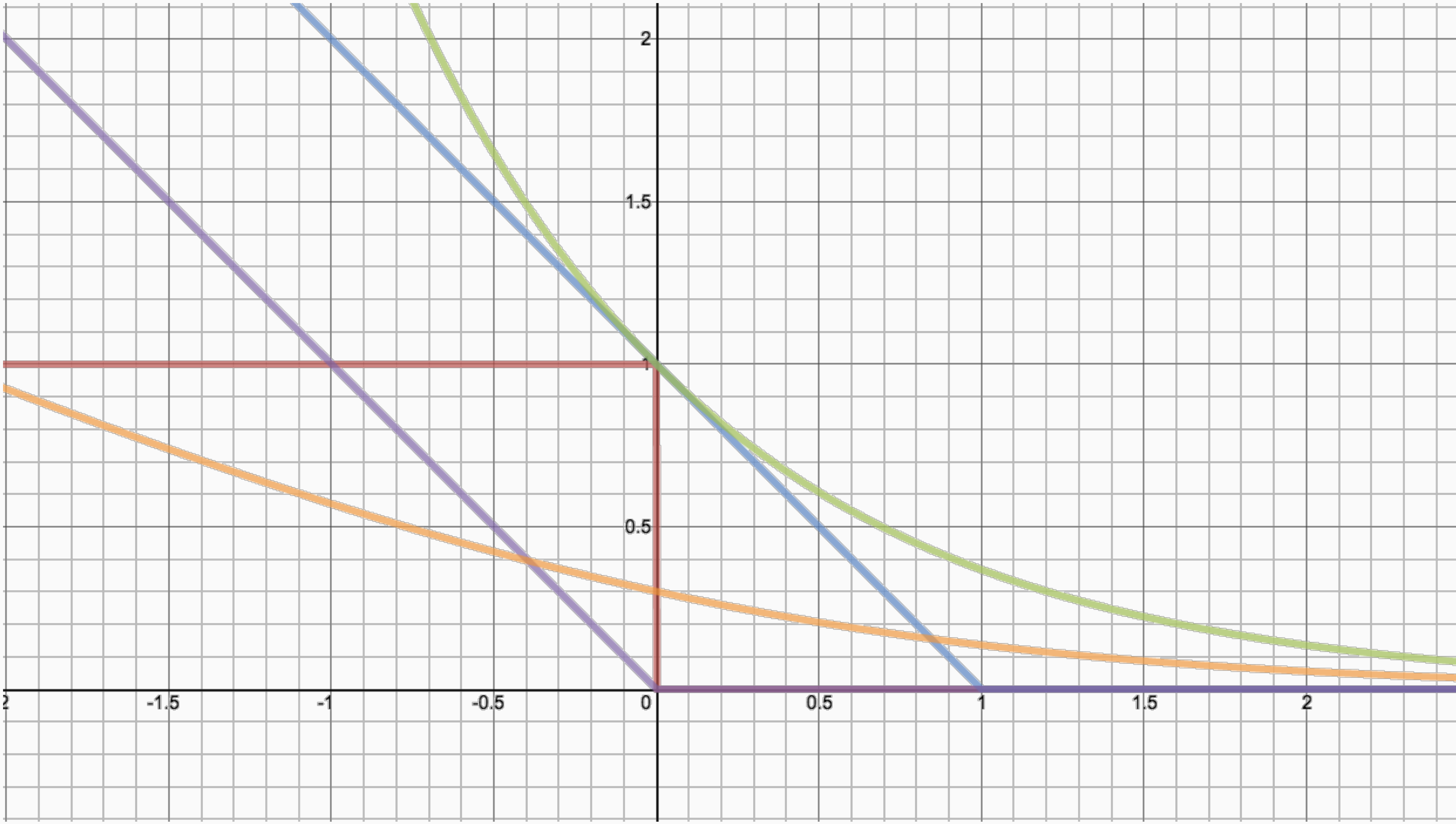
The loss function zoo

Many loss functions exist

- Perceptron loss $$L_{Perceptron}(y, \mathbf{x}, \mathbf{w}) = \max(0, -y\mathbf{w}^T\mathbf{x})$$
- Hinge loss (SVM) $$L_{Hinge}(y, \mathbf{x}, \mathbf{w}) = \max(0, 1 - y\mathbf{w}^T\mathbf{x})$$
- Exponential loss (AdaBoost) $$L_{Exponential}(y, \mathbf{x}, \mathbf{w}) = e^{-y\mathbf{w}^T\mathbf{x}}$$
- Logistic loss (logistic regression) $$L_{Logistic}(y, \mathbf{x}, \mathbf{w}) = \log(1 + e^{-y\mathbf{w}^T\mathbf{x}})$$

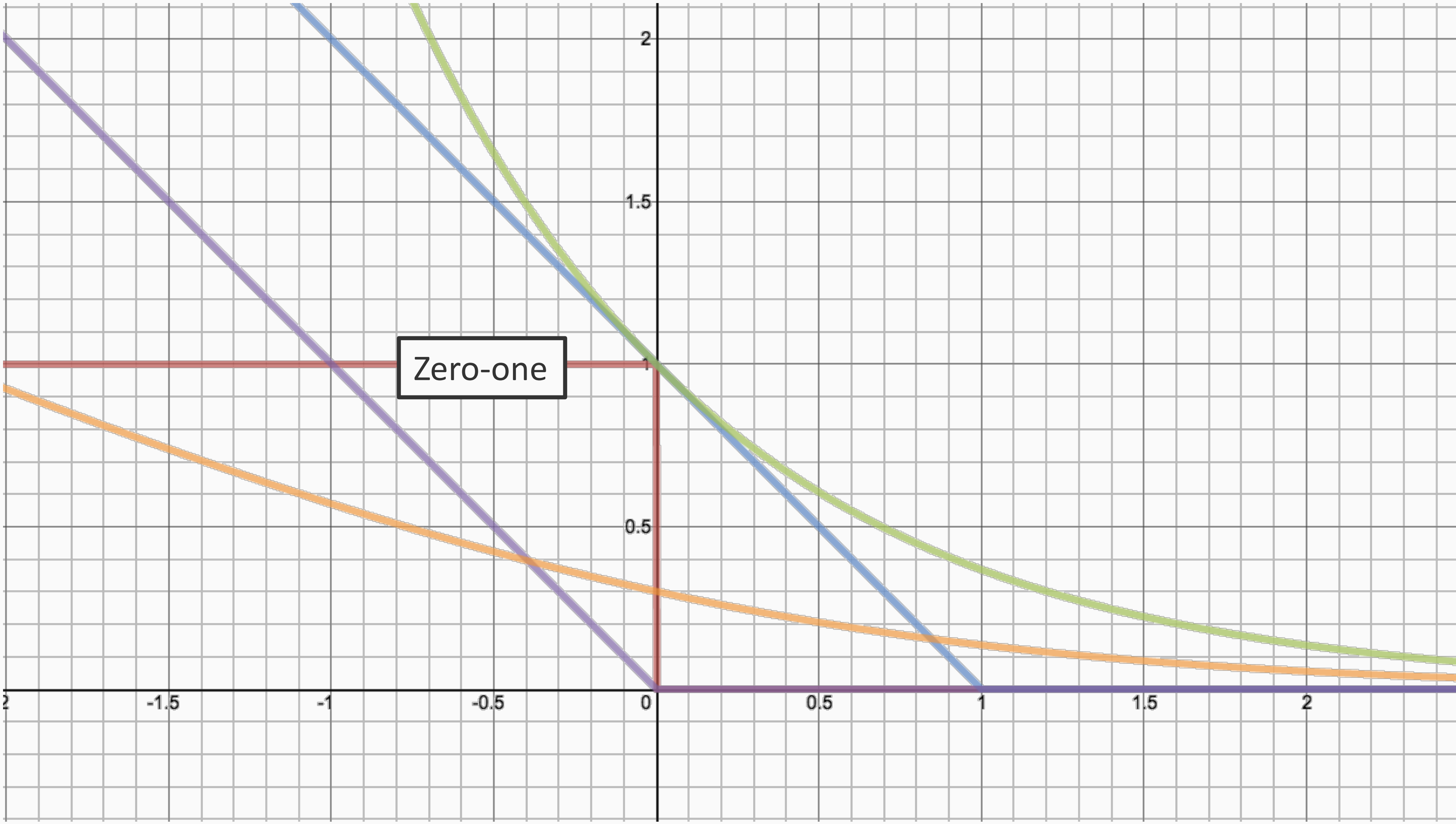
$$\min_{h \in H} \text{regularizer}(h) + C\frac{1}{m}\sum_i L(h(\mathbf{x}_i), f(\mathbf{x}_i))$$

The loss function zoo



$$\min_{h \in H} \text{regularizer}(h) + C\frac{1}{m}\sum_i L(h(\mathbf{x}_i), f(\mathbf{x}_i))$$

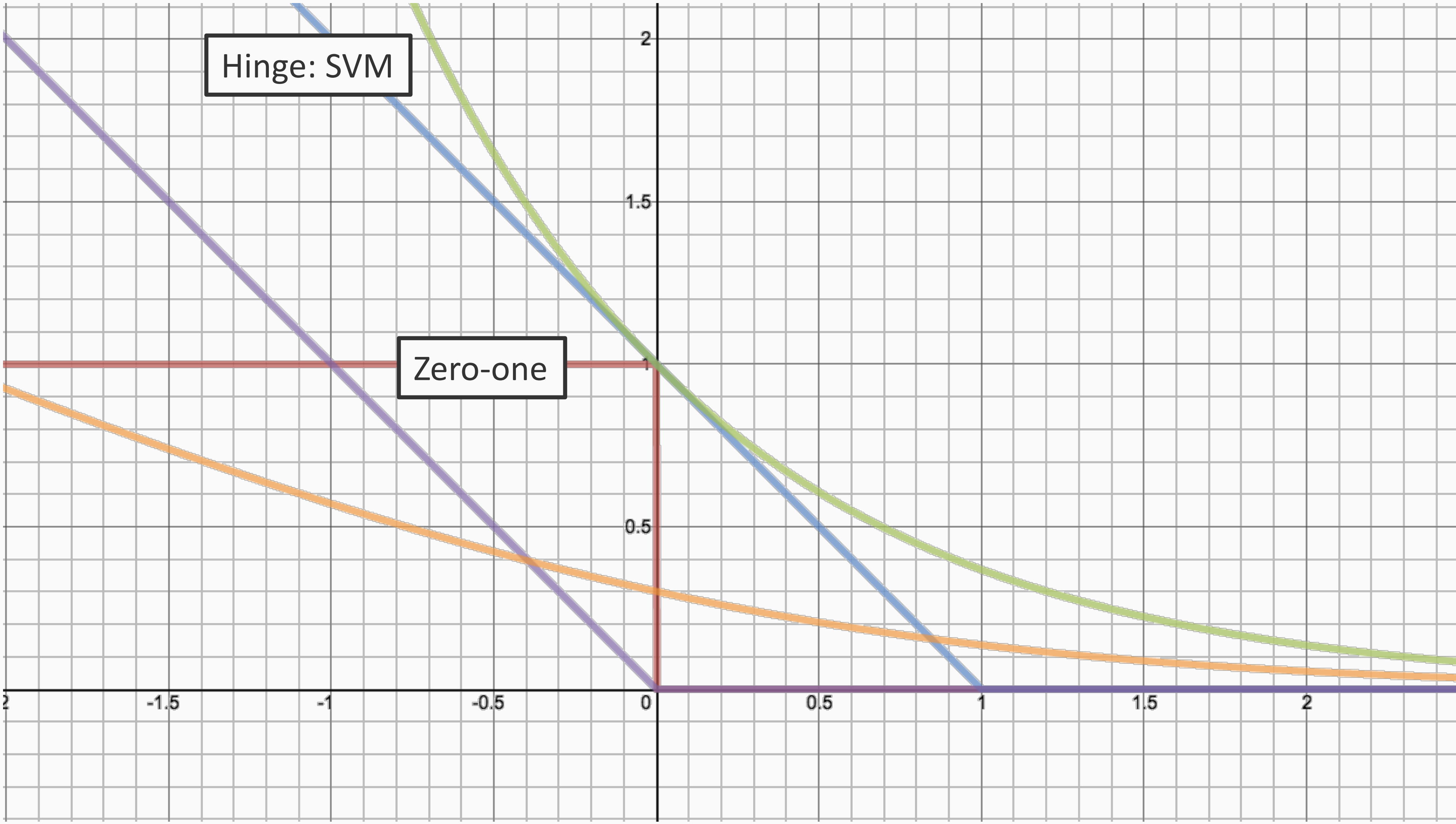
The loss function zoo



$$\min_{h \in H} \text{regularizer}(h) + C\frac{1}{m}\sum_i L(h(\mathbf{x}_i), f(\mathbf{x}_i))$$

The loss function zoo

$$L_{Hinge}(y, \mathbf{x}, \mathbf{w}) = \max(0, 1 - y\mathbf{w}^T\mathbf{x})$$

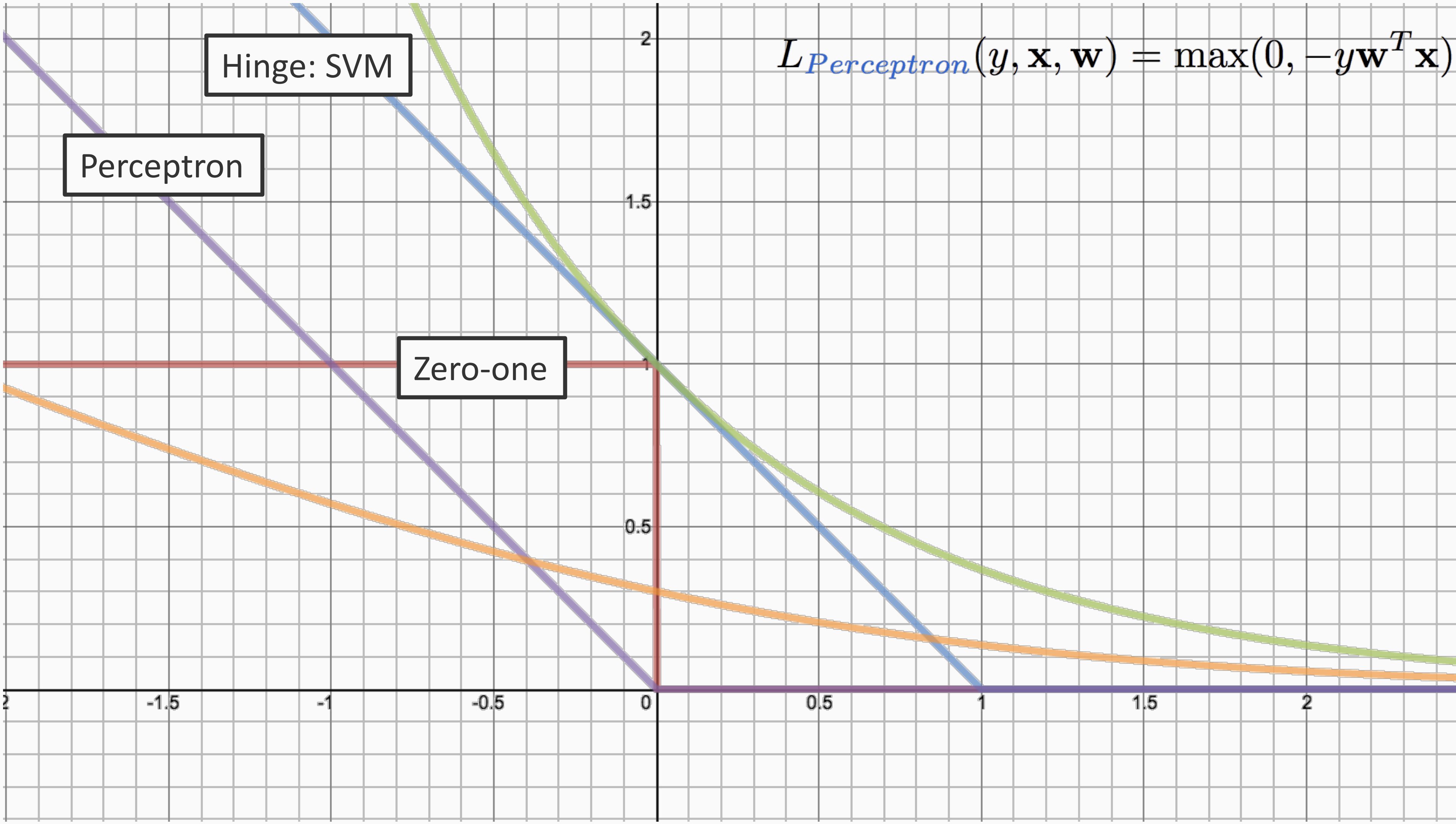


$$\min_{h \in H} \text{regularizer}(h) + C\frac{1}{m}\sum_i L(h(\mathbf{x}_i), f(\mathbf{x}_i))$$

The loss function zoo

$$L_{Hinge}(y, \mathbf{x}, \mathbf{w}) = \max(0, 1 - y\mathbf{w}^T\mathbf{x})$$

$$L_{Perceptron}(y, \mathbf{x}, \mathbf{w}) = \max(0, -y\mathbf{w}^T\mathbf{x})$$



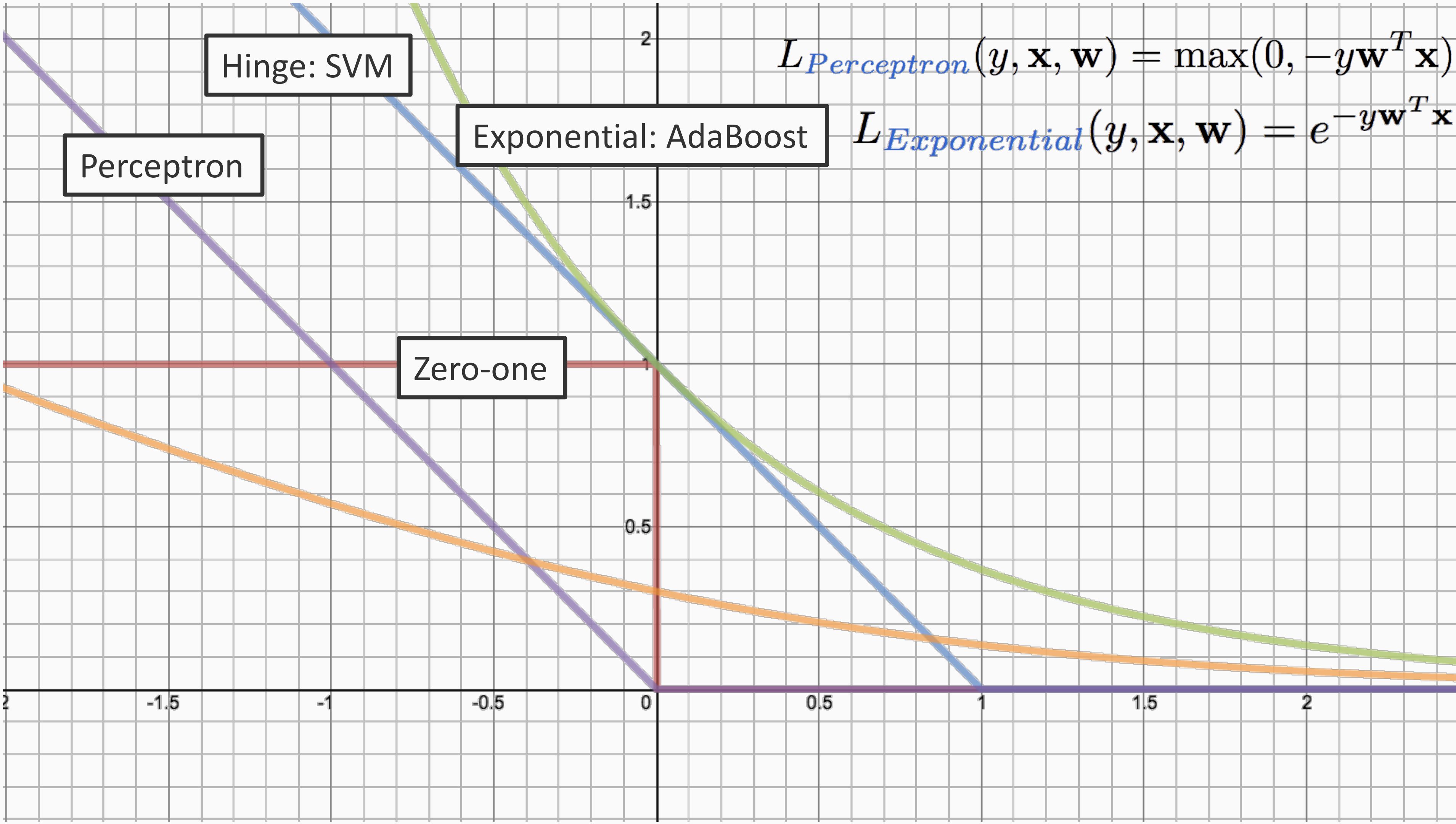
$$\min_{h \in H} \text{regularizer}(h) + C\frac{1}{m}\sum_i L(h(\mathbf{x}_i), f(\mathbf{x}_i))$$

The loss function zoo

$$L_{Hinge}(y, \mathbf{x}, \mathbf{w}) = \max(0, 1 - y\mathbf{w}^T\mathbf{x})$$

$$L_{Perceptron}(y, \mathbf{x}, \mathbf{w}) = \max(0, -y\mathbf{w}^T\mathbf{x})$$

$$L_{Exponential}(y, \mathbf{x}, \mathbf{w}) = e^{-y\mathbf{w}^T\mathbf{x}}$$



$$\min_{h \in H} \text{regularizer}(h) + C\frac{1}{m}\sum_i L(h(\mathbf{x}_i), f(\mathbf{x}_i))$$

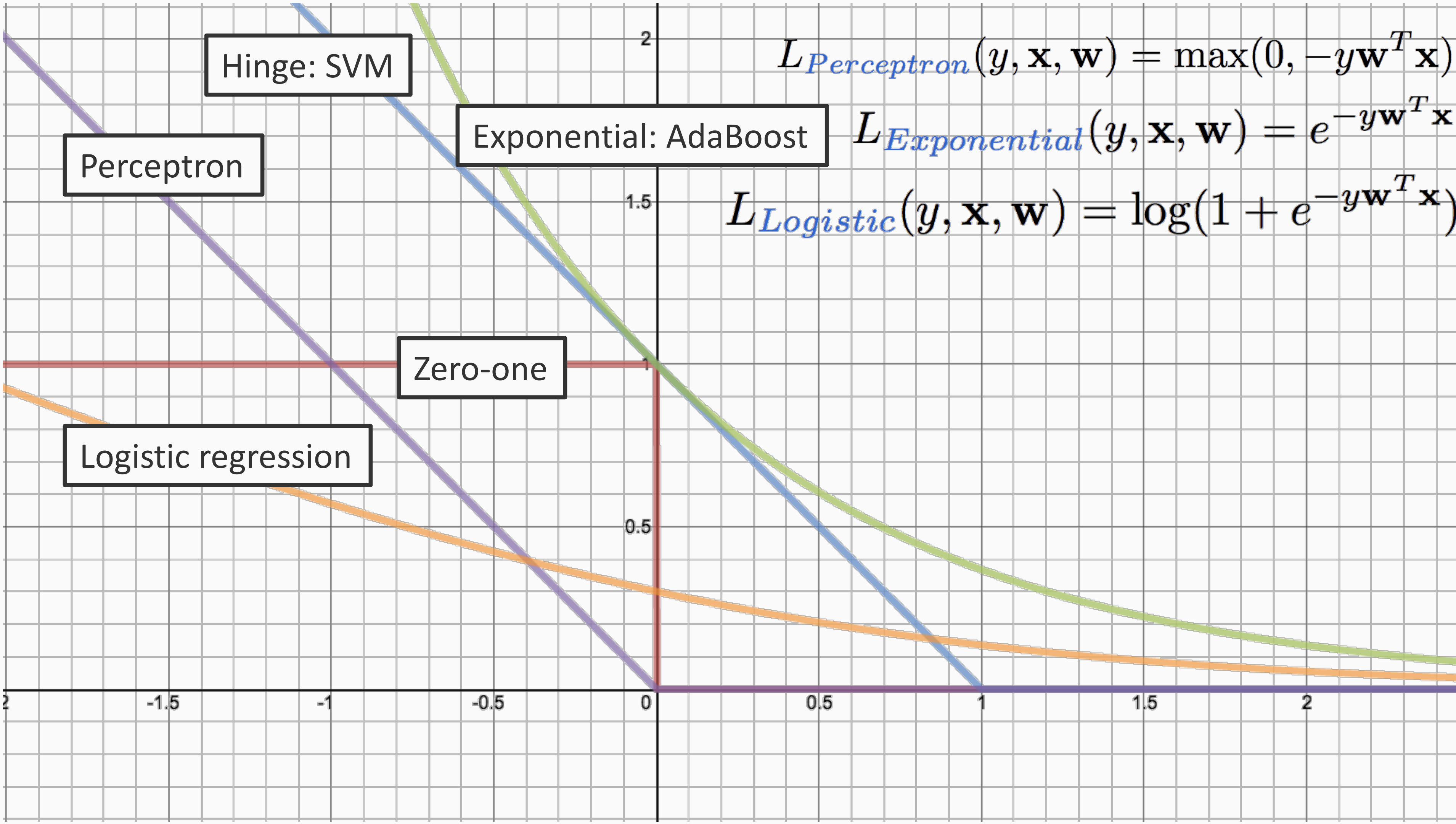
The loss function zoo

$$L_{Hinge}(y, \mathbf{x}, \mathbf{w}) = \max(0, 1 - y\mathbf{w}^T\mathbf{x})$$

$$L_{Perceptron}(y, \mathbf{x}, \mathbf{w}) = \max(0, -y\mathbf{w}^T\mathbf{x})$$

$$L_{Exponential}(y, \mathbf{x}, \mathbf{w}) = e^{-y\mathbf{w}^T\mathbf{x}}$$

$$L_{Logistic}(y, \mathbf{x}, \mathbf{w}) = \log(1 + e^{-y\mathbf{w}^T\mathbf{x}})$$



The loss function zoo

Zoomed out

The loss function zoo

Zoomed out even more