

is for classification

Lecture 4: Logistic Regression

Machine Learning, Summer Term 2019

Michael Tangemann Frank Hutter Marius Lindauer

University of Freiburg



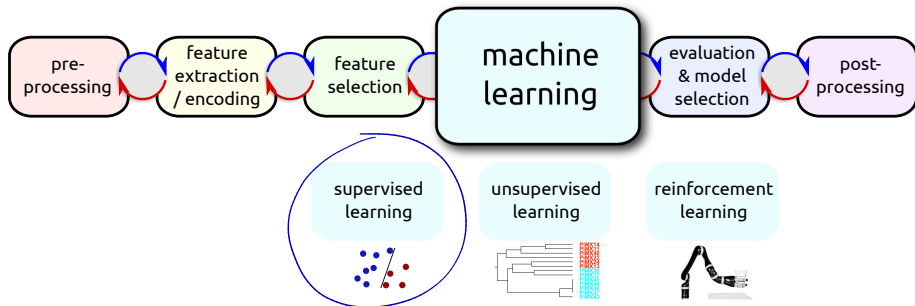
Lecture Overview

- 1 Motivation
- 2 The Model
- 3 How to Derive the Parameters...
- 4 Wrapup: Summary, Related Topics, Preview

Lecture Overview

- 1 Motivation
- 2 The Model
- 3 How to Derive the Parameters...
- 4 Wrapup: Summary, Related Topics, Preview

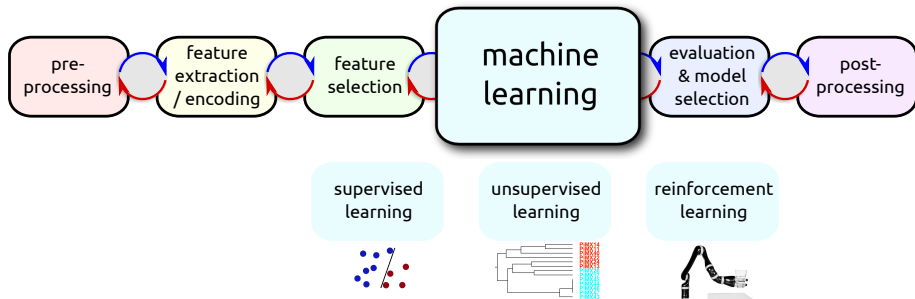
ML Design Cycle



Today's topic is **classification**:

- Use past experience to predict the future

ML Design Cycle

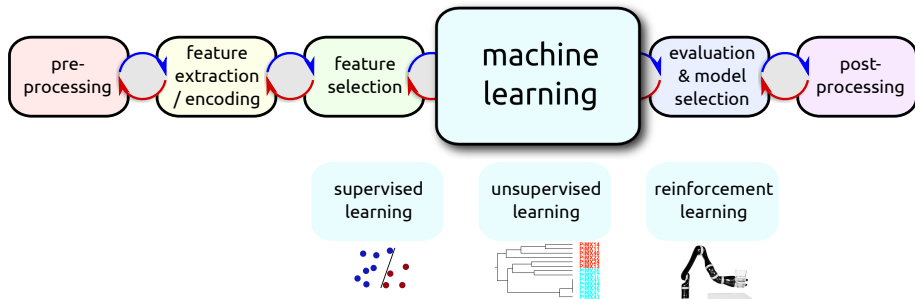


Today's topic is **classification**:

- Use past experience to predict the future
- Use labelled data points $\langle (\mathbf{x}_i, y_i) \rangle_{i=1}^N$

features $\mathbf{x}_i \in \mathbb{R}^D$
labels $y_i \in \{0, 1\}$

ML Design Cycle



Today's topic is **classification**:

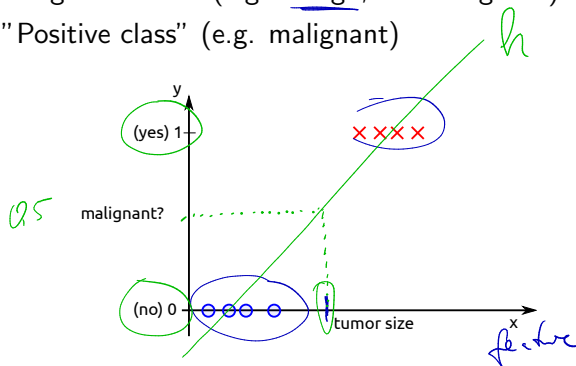
- Use past experience to predict the future
- Use labelled data points $\langle (\mathbf{x}_i, y_i) \rangle_{i=1}^N$
- Train a **model** which can predict the label y_{N+1} of a new data point \mathbf{x}_{N+1}

Simple Example: Tumor Classification

Is the tumor benign or malignant?

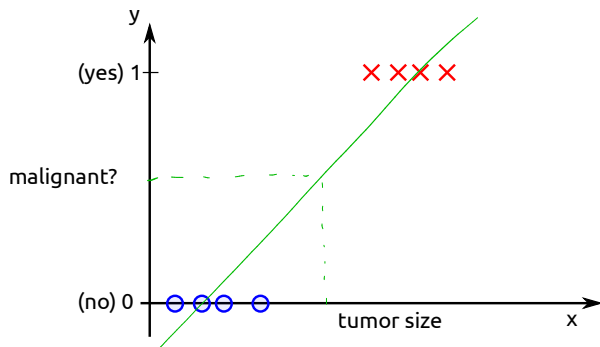
Labels $y \in \{0, 1\}$, with

- $y = 0$: "Negative class" (e.g. benign, not malignant)
- $y = 1$: "Positive class" (e.g. malignant)



Try fitting a (augmented) model $h_{\mathbf{w}}(\mathbf{x}) = \underline{\mathbf{w}}^T \mathbf{x}$ as in linear regression...

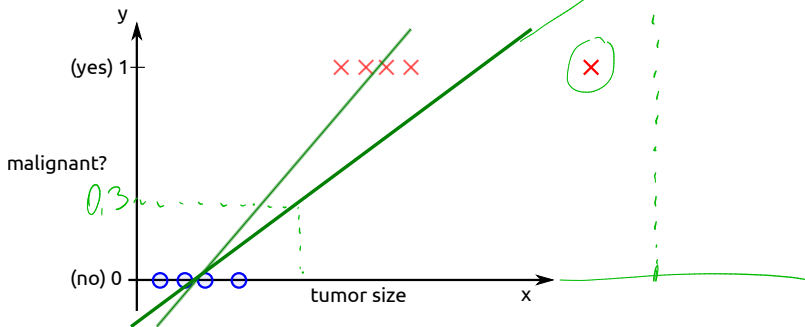
Simple Example: Tumor Classification



Threshold the classifier output $h_{\mathbf{w}}(\mathbf{x})$ at 0.5:

- If $h_{\mathbf{w}}(\mathbf{x}) > 0.5$, predict $y = 1$: (malignant)
- If $h_{\mathbf{w}}(\mathbf{x}) \leq 0.5$, predict $y = 0$: (not malignant)

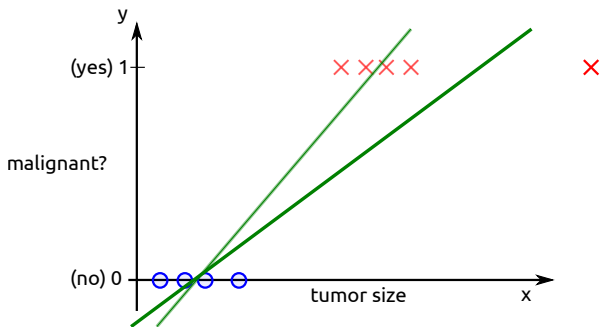
Simple Example: Tumor Classification



Outliers? Houston, we have a problem...

- For classification case we should only accept $y = 0$ or $y = 1$.
- However, linear regression model $h_{\mathbf{w}}(\mathbf{x})$ can generate $y > 1$ or $y < 0$.

Simple Example: Tumor Classification



Outliers? Houston, we have a problem...

- For classification case we should only accept $y = 0$ or $y = 1$.
- However, linear regression model $h_{\mathbf{w}}(\mathbf{x})$ can generate $y > 1$ or $y < 0$.

Solution: [logistic regression](#), which bounds the output to $0 \leq h_{\mathbf{w}}(\mathbf{x}) \leq 1$

Lecture Overview

- 1 Motivation
- 2 The Model**
- 3 How to Derive the Parameters...
- 4 Wrapup: Summary, Related Topics, Preview

Logistic Regression Model

We want: $0 \leq \underline{h_{\mathbf{w}}(\mathbf{x})} \leq 1$

augmented notation!

Standard linear regression
(using the inner product) delivers:

$$h_{\mathbf{w}}(\mathbf{x}) = \mathbf{w}^T \underline{\mathbf{x}}$$

Logistic Regression Model

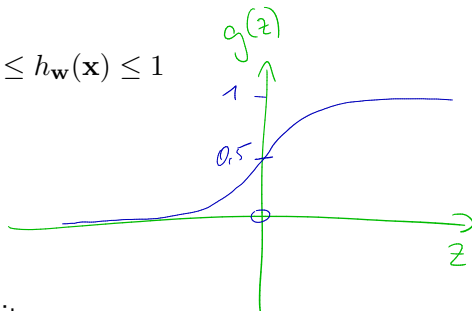
We want: $0 \leq h_{\mathbf{w}}(\mathbf{x}) \leq 1$

Standard linear regression
(using the inner product) delivers:

$$h_{\mathbf{w}}(\mathbf{x}) = \mathbf{w}^T \mathbf{x}$$

A subtle change introduces non-linearity:

$$h_{\mathbf{w}}(\mathbf{x}) = g(\mathbf{w}^T \mathbf{x}) \text{ with } g(z) = \frac{1}{1+e^{-z}}$$



Logistic Regression Model

We want: $0 \leq h_{\mathbf{w}}(\mathbf{x}) \leq 1$

Standard linear regression
(using the inner product) delivers:

$$h_{\mathbf{w}}(\mathbf{x}) = \mathbf{w}^T \mathbf{x}$$

A subtle change introduces non-linearity:

$$h_{\mathbf{w}}(\mathbf{x}) = g(\mathbf{w}^T \mathbf{x}) \text{ with } g(z) = \frac{1}{1+e^{-z}}$$

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

Logistic Regression Model

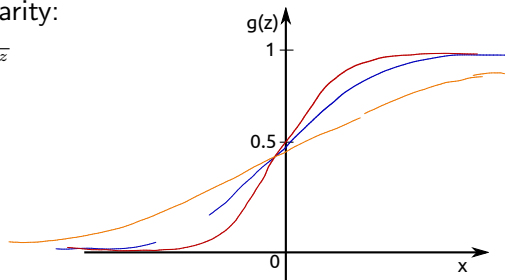
We want: $0 \leq h_{\mathbf{w}}(\mathbf{x}) \leq 1$

Standard linear regression
(using the inner product) delivers:
$$h_{\mathbf{w}}(\mathbf{x}) = \mathbf{w}^T \mathbf{x}$$

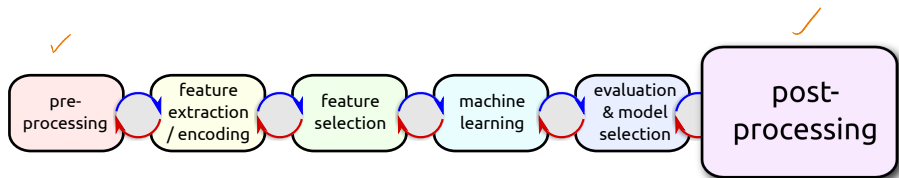
$g(z)$ is called **sigmoid function**
or **logistic function**

A subtle change introduces non-linearity:
$$h_{\mathbf{w}}(\mathbf{x}) = g(\mathbf{w}^T \mathbf{x}) \text{ with } g(z) = \frac{1}{1+e^{-z}}$$

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



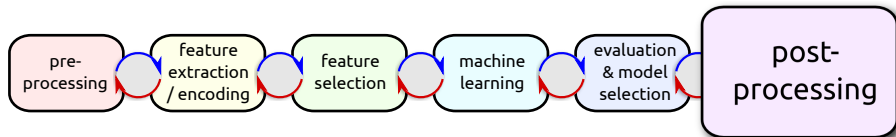
Role of Non-linear Function g



Please observe:

- The use of function $g(\mathbf{x})$ has similarity to a post-processing of a linear method

Role of Non-linear Function g



Please observe:

- The use of function $g(\mathbf{x})$ has similarity to a post-processing of a linear method
- You can actually apply a post processing to linear classifier outputs by
 - → **logistic calibration** (making distribution assumptions)
 - → **isotonic calibration** (no assumptions) *more data necessary*

Interpretation of Model Output

$h_{\mathbf{w}}(\mathbf{x}) =$ estimated probability, that $y = 1$ on input \mathbf{x}

Example with tumor size:

$$x = \begin{bmatrix} \mathbf{x}_0 \\ \mathbf{x}_1 \end{bmatrix} = \begin{bmatrix} 1 \\ tumorSize \end{bmatrix}$$

Interpretation of Model Output

$h_{\mathbf{w}}(\mathbf{x}) =$ estimated probability, that $y = 1$ on input \mathbf{x}

Example with tumor size:

$$x = \begin{bmatrix} \mathbf{x}_0 \\ \mathbf{x}_1 \end{bmatrix} = \begin{bmatrix} 1 \\ tumorSize \end{bmatrix}$$

How can we interpret the model output $h_{\mathbf{w}}(\mathbf{x}) = 0.7?$

→ We would infer that with a probability of 70% the patient's tumor is malignant.

Interpretation of Model Output

$f(\dots)$

$h_{\mathbf{w}}(\mathbf{x}) \approx$ estimated probability, that $y = 1$

More formally, we work with a *hypothesis*:

Interpretation of Model Output

$h_{\mathbf{w}}(\mathbf{x}) \approx$ estimated probability, that $y = 1$

More formally, we work with a *hypothesis*:

$$h_{\mathbf{w}}(\mathbf{x}) = P(\underline{y = 1} | \mathbf{x}; \mathbf{w})$$

data / feature
↓
model →

"probability that $y = 1$,
given that the input is \mathbf{x} and
the model is parameterized by \mathbf{w} "

Interpretation of Model Output

$h_{\mathbf{w}}(\mathbf{x}) \approx$ estimated probability, that $y = 1$

More formally, we work with a *hypothesis*:

$$h_{\mathbf{w}}(\mathbf{x}) = P(y = 1 | \mathbf{x}; \mathbf{w})$$

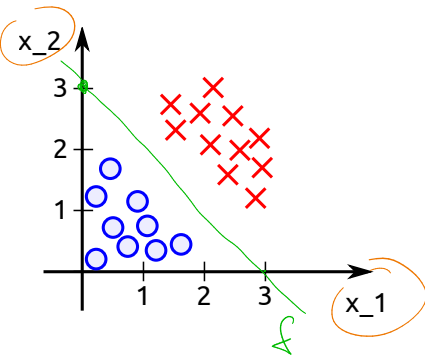
"probability that $y = 1$,
given that the input is \mathbf{x} and
the model is parameterized by \mathbf{w} "

The actual labels are still discrete
($y = 0$ or $y = 1$), but the
probabilities need to add to one:

$$P(y = 0 | \mathbf{x}; \mathbf{w}) + P(y = 1 | \mathbf{x}; \mathbf{w}) = 1$$
$$P(y = 0 | \mathbf{x}; \mathbf{w}) = 1 - P(y = 1 | \mathbf{x}; \mathbf{w})$$

Example in \mathbb{R}^2

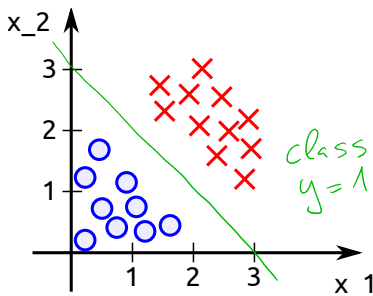
Where is the decision boundary?



$$h_{\mathbf{w}}(\mathbf{x}) = g(\mathbf{w}_0 + \mathbf{w}_1 x_1 + \mathbf{w}_2 x_2)$$

Example in \mathbb{R}^2

Where is the decision boundary?



$$h_{\mathbf{w}}(\mathbf{x}) = g(\mathbf{w}_0 + \mathbf{w}_1 x_1 + \mathbf{w}_2 x_2)$$

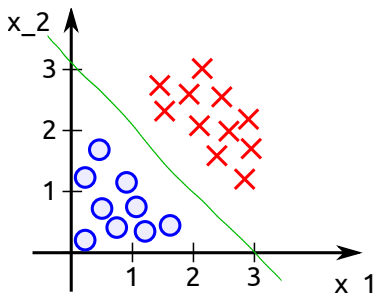
$$\underline{\mathbf{w}} = \begin{bmatrix} -3 \\ 1 \\ 1 \end{bmatrix}$$

Predict " $y = 1$ " if

$$\underline{-3 + x_1 + x_2 \geq 0}$$

Example in \mathbb{R}^2

Where is the decision boundary?



$$h_{\mathbf{w}}(\mathbf{x}) = g(\mathbf{w}_0 + \mathbf{w}_1 \mathbf{x}_1 + \mathbf{w}_2 \mathbf{x}_2)$$

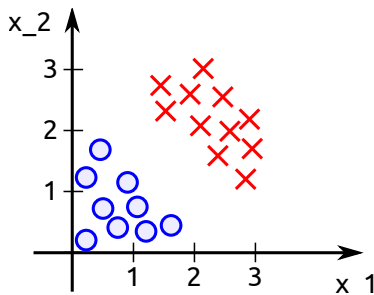
$$\mathbf{w} = \begin{bmatrix} -3 \\ 1 \\ 1 \end{bmatrix}$$

Predict " $y = 1$ " if $-3 + \mathbf{x}_1 + \mathbf{x}_2 \geq 0$

$$\boxed{\mathbf{x}_1 + \mathbf{x}_2 = 3} \Leftrightarrow \underline{h_{\mathbf{w}}(\mathbf{x}) = 0.5}$$

Example in \mathbb{R}^2

Where is the decision boundary?



$$h_{\mathbf{w}}(\mathbf{x}) = g(\mathbf{w}_0 + \mathbf{w}_1 \mathbf{x}_1 + \mathbf{w}_2 \mathbf{x}_2)$$

$$\mathbf{w} = \begin{bmatrix} -3 \\ 1 \\ 1 \end{bmatrix}$$

Predict " $y = 1$ " if $-3 + \mathbf{x}_1 + \mathbf{x}_2 \geq 0$

$$\mathbf{x}_1 + \mathbf{x}_2 = 3 \Leftrightarrow h_{\mathbf{w}}(\mathbf{x}) = 0.5$$

Remark: as in linear models, the use of additional dimensions and basis functions allows for non-linear decision boundaries!

Lecture Overview

- 1 Motivation
- 2 The Model
- 3 How to Derive the Parameters...**
- 4 Wrapup: Summary, Related Topics, Preview

How to Choose the Weights / Parameters \mathbf{w} ?

We assume labelled training data points $\langle (\mathbf{x}_i, y_i) \rangle_{i=1}^N$


How to Choose the Weights / Parameters \mathbf{w} ?

We assume labelled training data points $\langle (\mathbf{x}_i, y_i) \rangle_{i=1}^N$

- If we use the augmented notation – what is the dimensionality of the data? 🙌 🙌



How to Choose the Weights / Parameters \mathbf{w} ?

We assume labelled training data points $\langle (\mathbf{x}_i, y_i) \rangle_{i=1}^N$

- If we use the augmented notation – what is the dimensionality of the data? 
- Answer: $\mathbf{x} \in \mathbb{R}^{D+1}$ with first dimension $x_0 = 1$

How to Choose the Weights / Parameters \mathbf{w} ?

We assume labelled training data points $\langle (\mathbf{x}_i, y_i) \rangle_{i=1}^N$

- If we use the augmented notation – what is the dimensionality of the data? 
- Answer: $\mathbf{x} \in \mathbb{R}^{D+1}$ with first dimension $x_0 = 1$ 

Our model is:

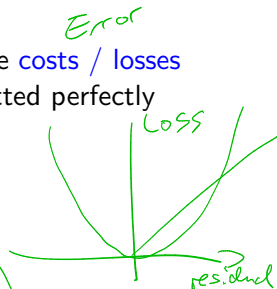
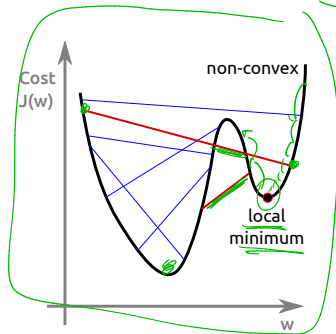
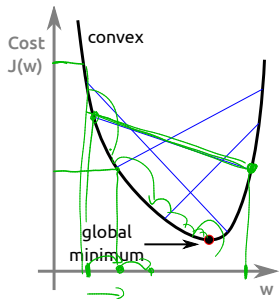
$$h_{\mathbf{w}}(\mathbf{x}) = \frac{1}{1 + e^{-\mathbf{w}^T \mathbf{x}}}$$

How can we choose the parameters \mathbf{w} ?

Reminder: Loss Function Punishes Wrong Predictions

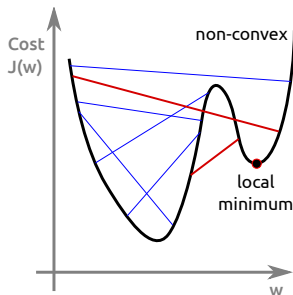
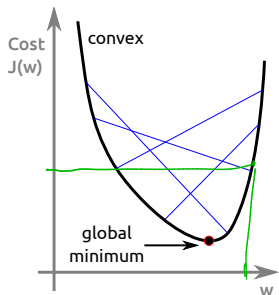
- To tune the weight vector w , we should check the costs / losses generated by data points, which have not been fitted perfectly (and thus show non-zero residuals).

gradient descent



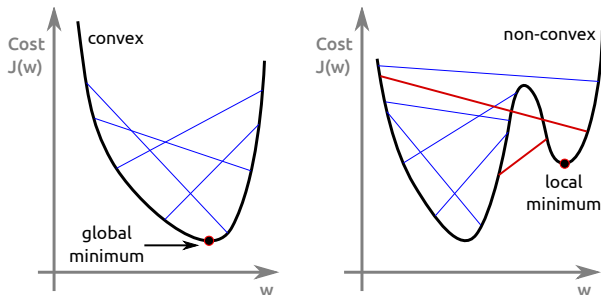
Reminder: Loss Function Punishes Wrong Predictions

- To tune the weight vector \mathbf{w} , we should check the **costs / losses** generated by data points, which have not been fitted perfectly (and thus show non-zero residuals).
- Loss for data point (\mathbf{x}, y) depends on the choice of \mathbf{w} : $Cost(h_{\mathbf{w}}, y)$.



Reminder: Loss Function Punishes Wrong Predictions

- To tune the weight vector \mathbf{w} , we should check the **costs / losses** generated by data points, which have not been fitted perfectly (and thus show non-zero residuals).
- Loss for data point (\mathbf{x}, y) depends on the choice of \mathbf{w} : $Cost(h_{\mathbf{w}}, y)$.
- Parameters w_i can be optimized easier, if the loss function is **convex** and (continuously) **differentiable**.



Quadratic Loss is Unfavorable for Logistic Regression

Unfortunately, a quadratic loss function (as e.g. in LDA)

$$\underline{J(\mathbf{w})} = ||h_{\mathbf{w}}(\mathbf{x}), y||^2$$

would lead to a non-convex optimization problem with local minima.

- What may cause this trouble?



Quadratic Loss is Unfavorable for Logistic Regression

Unfortunately, a quadratic loss function (as e.g. in LDA)

$$J(\mathbf{w}) = \|h_{\mathbf{w}}(\mathbf{x}), y\|^2$$

would lead to a non-convex optimization problem with local minima.

- What may cause this trouble?



- Answer: the sigmoid function in $h_{\mathbf{w}}(\mathbf{x})$

Quadratic Loss is Unfavorable for Logistic Regression

Unfortunately, a quadratic loss function (as e.g. in LDA)

$$J(\mathbf{w}) = \|h_{\mathbf{w}}(\mathbf{x}), y\|^2$$

would lead to a non-convex optimization problem with local minima.

- What may cause this trouble?



- Answer: the sigmoid function in $h_{\mathbf{w}}(\mathbf{x})$

- How can we derive better loss functions?



Quadratic Loss is Unfavorable for Logistic Regression

Unfortunately, a quadratic loss function (as e.g. in LDA)

$$J(\mathbf{w}) = ||h_{\mathbf{w}}(\mathbf{x}), y||^2$$

would lead to a non-convex optimization problem with local minima.

- What may cause this trouble?



- Answer: the sigmoid function in $h_{\mathbf{w}}(\mathbf{x})$

- How can we derive better loss functions?

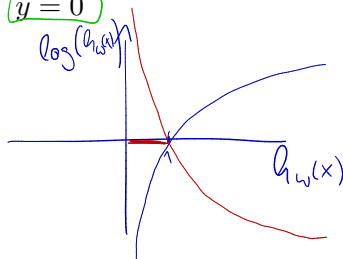


- Answer: talk to domain experts!

Adapted Loss Function of Logistic Regression (I)

Proposed loss function (specifically adapted to logistic regression):

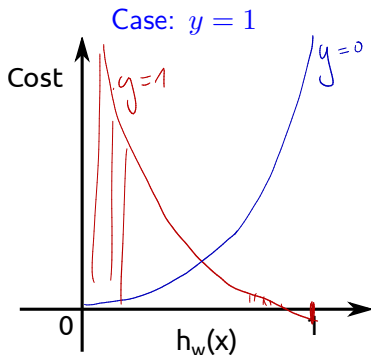
$$J(h_{\mathbf{w}}(\mathbf{x}), y) = \begin{cases} -\log(h_{\mathbf{w}}(\mathbf{x})) & \text{for } y = 1 \text{ (malignant)} \\ -\log(1 - h_{\mathbf{w}}(\mathbf{x})) & \text{for } y = 0 \end{cases}$$



Adapted Loss Function of Logistic Regression (I)

Proposed loss function (specifically adapted to logistic regression):

$$J(h_{\mathbf{w}}(\mathbf{x}), y) = \begin{cases} -\log(h_{\mathbf{w}}(\mathbf{x})) & \text{for } y = 1 \text{ (malignant)} \\ -\log(1 - h_{\mathbf{w}}(\mathbf{x})) & \text{for } y = 0 \end{cases}$$

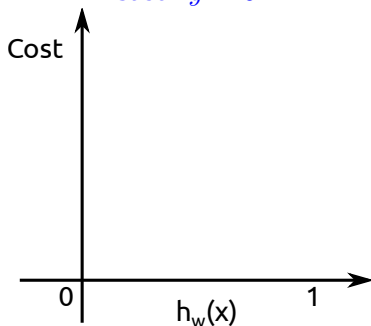


Adapted Loss Function of Logistic Regression (I)

Proposed cost function for the logistic regression model:

$$\text{Cost}(h_{\mathbf{w}}(\mathbf{x}), y) = \begin{cases} -\log(h_{\mathbf{w}}(\mathbf{x})) & \text{for } y = 1 \\ -\log(1 - h_{\mathbf{w}}(\mathbf{x})) & \text{for } y = 0 \end{cases}$$

Case: $y = 0$



Adapted Loss Function for Logistic Regression (II)

The case distinction can be avoided using this formulation (check by setting $y = 0$ and $y = 1$):

$$J(h_{\mathbf{w}}(\mathbf{x}), y) = -y \log(h_{\mathbf{w}}(\mathbf{x})) - ((1 - y) \log(1 - h_{\mathbf{w}}(\mathbf{x})))$$

- Nice property: J is **convex**.
- Not so nice property: There is no analytic solution (but gradient descent works well).

Literature:

- Big parts of this section are based on the lectures by Andrew Ng (see youtube channel) on logistic regression.
- For a good reading on logistic calibration of linear classifiers, see Section 7.4 (Obtaining probabilities from linear classifiers) in Peter Flach: Machine Learning (Cambridge Univ. Press, 2012)

For Toolbox Use: Check Definition of Labels

For the assignment, you have used / will use the scikit-learn toolbox. Please note, that this toolbox uses labels $y \in \{-1, 1\}$. By reformulation, this leads to the following alternative loss function for logistic regression:

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

Again, check by setting the labels to fixed values and compare the both formulations:

- Case 1: $y = 0$ (our cost function / Andrew Ng) and $y = -1$ (scikit-learn)
- Case 1: $y = 1$ (our cost function / Andrew Ng) and $y = 1$ (scikit-learn)

Lecture Overview

- 1 Motivation
- 2 The Model
- 3 How to Derive the Parameters...
- 4 **Wrapup: Summary, Related Topics, Preview**

Summary by learning goals

Having heard this lecture, you can now ...

- explain, why probability outputs instead of class values may be desired
- describe the logistic function
- formulate the logistic regression hypothesis
- formulate the optimization criterion and explain, how parameters are determined
- explain, how the output of logistic regression is interpreted
- (assignments) formulate pros and cons of LDA and logistic regression
- (assignments) compare assumptions and the effects of their violation for the two classification approaches

