Linear classification
Motivation
Motivation

- Labels in classification problems don’t quite “fit” linear regression models

For classification, error rate is the primary benchmark (rather than mean squared error)

- Can the benefits of linearity be enjoyed for classification?
Logistic Regression
Logistic regression model:
Training data $(\vec{X}_1, Y_1), \ldots, (\vec{X}_n, Y_n)$ and “future” example $(\vec{X}, Y)$ are IID, and for any $\vec{x} \in \mathbb{R}^d$,

$$(Y \mid \vec{X} = \vec{x}) \sim \text{Bernoulli}(\text{logistic}(\vec{x} \cdot \vec{w}))$$
Linear model for binary classification

Logistic regression model:
Training data \((\vec{X}_1, Y_1), \ldots, (\vec{X}_n, Y_n)\) and “future” example \((\vec{X}, Y)\) are IID, and for any \(\vec{x} \in \mathbb{R}^d\),

\[
(Y \mid \vec{X} = \vec{x}) \sim \text{Bernoulli}(\text{logistic}(\vec{x} \cdot \vec{w}))
\]

- \(\text{logistic}(t) := \frac{e^t}{1 + e^t} \in (0, 1)\) for \(t \in \mathbb{R}\)
- \(\vec{w} \in \mathbb{R}^d\) is a parameter of the model
- Marginal distribution of \(\vec{X}\) unspecified; does not depend on \(\vec{w}\).
Log odds ratio and linear classifiers

In logistic regression model, the log odds ratio (a.k.a. logit function) is a linear function:

\[
\ln \frac{\Pr(Y = 1 \mid \vec{X} = \vec{x})}{\Pr(Y = 0 \mid \vec{X} = \vec{x})} = \ln e^{\vec{x} \cdot \vec{w}} = \vec{x} \cdot \vec{w}
\]

(Negating the log odds ratio just swaps the 1 and 0 labels on left-hand side)
In logistic regression model, the log odds ratio (a.k.a. logit function) is a linear function:

\[
\ln \frac{\Pr(Y = 1 \mid \vec{X} = \vec{x})}{\Pr(Y = 0 \mid \vec{X} = \vec{x})} = \ln e^{\vec{x} \cdot \vec{w}} = \vec{x} \cdot \vec{w}
\]

(Negating the log odds ratio just swaps the 1 and 0 labels on left-hand side)

**Observation:** Given \( \vec{X} = \vec{x} \), label 1 is more likely than label 0 if and only if \( \vec{x} \cdot \vec{w} > 0 \)
Log odds ratio and linear classifiers

In logistic regression model, the **log odds ratio** (a.k.a. **logit** function) is a linear function:

\[
\ln \frac{\Pr(Y = 1 \mid \vec{X} = \vec{x})}{\Pr(Y = 0 \mid \vec{X} = \vec{x})} = \ln e^{\vec{x} \cdot \vec{w}} = \vec{x} \cdot \vec{w}
\]

(Negating the log odds ratio just swaps the 1 and 0 labels on left-hand side)

**Observation:** Given \(\vec{X} = \vec{x}\), label 1 is more likely than label 0 if and only if \(\vec{x} \cdot \vec{w} > 0\)

- In this model for specific parameter \(\vec{w}\), classifier with smallest error rate is

\[
f^*(\vec{x}) = 1 \{\vec{x} \cdot \vec{w} > 0\} = \begin{cases} 1 & \text{if } \vec{x} \cdot \vec{w} > 0 \\ 0 & \text{otherwise} \end{cases}
\]
In logistic regression model, the **log odds ratio** (a.k.a. **logit** function) is a linear function:

\[
\ln \frac{\Pr(Y = 1 \mid \vec{X} = \vec{x})}{\Pr(Y = 0 \mid \vec{X} = \vec{x})} = \ln e^{\vec{x} \cdot \vec{w}} = \vec{x} \cdot \vec{w}
\]

(Negating the log odds ratio just swaps the 1 and 0 labels on left-hand side)

**Observation:** Given $\vec{X} = \vec{x}$, label 1 is more likely than label 0 if and only if $\vec{x} \cdot \vec{w} > 0$

- In this model for specific parameter $\vec{w}$, classifier with smallest error rate is

\[
f^*(\vec{x}) = 1\{\vec{x} \cdot \vec{w} > 0\} = \begin{cases} 
1 & \text{if } \vec{x} \cdot \vec{w} > 0 \\
0 & \text{otherwise}
\end{cases}
\]

- If you believe this model *for some (unknown) parameter* $\vec{w}$, estimate $\vec{w}$ using training data!
Log odds ratio and linear classifiers

In logistic regression model, the **log odds ratio** (a.k.a. **logit** function) is a linear function:

$$
\ln \frac{\Pr(Y = 1 \mid \vec{X} = \vec{x})}{\Pr(Y = 0 \mid \vec{X} = \vec{x})} = \ln e^{\vec{x} \cdot \vec{w}} = \vec{x} \cdot \vec{w}
$$

(Negating the log odds ratio just swaps the 1 and 0 labels on left-hand side)

**Observation:** Given $\vec{X} = \vec{x}$, label 1 is more likely than label 0 if and only if $\vec{x} \cdot \vec{w} > 0$

- In this model for specific parameter $\vec{w}$, classifier with smallest error rate is

$$
f^* (\vec{x}) = 1 \{ \vec{x} \cdot \vec{w} > 0 \} = \begin{cases} 
1 & \text{if } \vec{x} \cdot \vec{w} > 0 \\
0 & \text{otherwise}
\end{cases}
$$

- If you believe this model *for some (unknown) parameter* $\vec{w}$, estimate $\vec{w}$ using training data!

Classifiers of the form $f(\vec{x}) = 1 \{ \vec{x} \cdot \vec{w} > 0 \}$ for some $\vec{w} \in \mathbb{R}^d$ are **linear classifiers**
Example #1: Classifying irises

- Two classes of irises: Setosa (0), Versicolor/Virginica (1)
- Two numerical features:
  - $x_1$: ratio of sepal width to sepal length
  - $x_2$: ratio of petal width to petal length
- 120 training data (40 from class 0, 80 from class 1)
Example #1: Logistic regression model for iris data

- Affine expansion: $\vec{x} = (x_1, x_2, 1)$
- Log odds ratio (with $\vec{w} \in \mathbb{R}^3$):
  \[ w_1 x_1 + w_2 x_2 + w_3 \]
- Linear classifier based on $\vec{w}$:
  \[ f(\vec{x}) = 1 \{ w_1 x_1 + w_2 x_2 + w_3 > 0 \} \]
Example #1: Logistic regression model for iris data

- **Affine expansion:** \( \vec{x} = (x_1, x_2, 1) \)
- **Log odds ratio (with \( \vec{w} \in \mathbb{R}^3 \)):**
  \[
  w_1 x_1 + w_2 x_2 + w_3
  \]
- **Linear classifier based on \( \vec{w} \):**
  \[
  f(\vec{x}) = \mathbb{1}\{w_1 x_1 + w_2 x_2 + w_3 > 0\}
  \]
- **Example:** \( \vec{w} = (-1, 0, -0.6) \)

\[
  f(\vec{x}) = \begin{cases} 
    1 & \text{if } x_1 < 0.6 \\
    0 & \text{if } x_1 \geq 0.6 
  \end{cases}
\]

(ignores \( x_2 \!\)!)
Example #1: Logistic regression model for iris data

- **Affine expansion:** \( \vec{x} = (x_1, x_2, 1) \)
- **Log odds ratio (with \( \vec{w} \in \mathbb{R}^3 \)):**
  \[ w_1 x_1 + w_2 x_2 + w_3 \]
- **Linear classifier based on \( \vec{w} \):**
  \[ f(\vec{x}) = \mathbb{1}\{w_1 x_1 + w_2 x_2 + w_3 > 0\} \]
- **Example:** \( \vec{w} = (-1, 0, -0.6) \)
  \[ f(\vec{x}) = \begin{cases} 1 & \text{if } x_1 < 0.6 \\ 0 & \text{if } x_1 \geq 0.6 \end{cases} \] (ignores \( x_2 \!\!\!\!\!\!\!\!\!\!\!\!\!\!\!\!\)!

In general, how can we estimate \( \vec{w} \) in the logistic regression model?
MLE for logistic regression model

Suppose training data $S$ are $(\vec{x}_1, y_1), \ldots, (\vec{x}_n, y_n)$, with $y_i \in \{0, 1\}$

**Recall:** Under logistic regression model with parameter $\vec{w}$,

$$(Y_i \mid \vec{X}_i = \vec{x}_i) \sim \text{Bernoulli}(\text{logistic}(\vec{x}_i \cdot \vec{w}))$$
MLE for logistic regression model

Suppose training data $S$ are $(\vec{x}_1, y_1), \ldots, (\vec{x}_n, y_n)$, with $y_i \in \{0, 1\}$

Recall: Under logistic regression model with parameter $\vec{w}$,

$$(Y_i \mid \vec{X}_i = \vec{x}_i) \sim \text{Bernoulli}(\text{logistic}(\vec{x}_i \cdot \vec{w}))$$

Likelihood of $\vec{w}$: Using fact that training examples are treated as IID,

$$L(\vec{w}) = p_{\vec{w}}((\vec{x}_1, y_1), \ldots, (\vec{x}_n, y_n)) \propto \prod_{i=1}^{n} \left\{ \begin{array}{ll} \text{logistic}(\vec{x}_i \cdot \vec{w}) & \text{if } y_i = 1 \\ 1 - \text{logistic}(\vec{x}_i \cdot \vec{w}) & \text{if } y_i = 0 \end{array} \right.$$  

(constant of proportionality doesn’t depend on $\vec{w}$)
MLE for logistic regression model

Suppose training data $S$ are $(\vec{x}_1, y_1), \ldots, (\vec{x}_n, y_n)$, with $y_i \in \{0, 1\}$

Recall: Under logistic regression model with parameter $\vec{w}$,

$$(Y_i \mid \vec{X}_i = \vec{x}_i) \sim \text{Bernoulli}(\text{logistic}(\vec{x}_i \cdot \vec{w}))$$

Likelihood of $\vec{w}$: Using fact that training examples are treated as IID,

$$L(\vec{w}) = p_{\vec{w}}((\vec{x}_1, y_1), \ldots, (\vec{x}_n, y_n)) \propto \prod_{i=1}^{n} \begin{cases} \text{logistic}(\vec{x}_i \cdot \vec{w}) & \text{if } y_i = 1 \\ 1 - \text{logistic}(\vec{x}_i \cdot \vec{w}) & \text{if } y_i = 0 \end{cases}$$

(constant of proportionality doesn’t depend on $\vec{w}$)

Log-likelihood of $\vec{w}$:

$$\ln L(\vec{w}) = -\sum_{i=1}^{n} \ln(1 + e^{\vec{x}_i \cdot \vec{w}}) + \sum_{i=1}^{n} y_i \vec{x}_i \cdot \vec{w} + \text{(terms that don’t depend on } \vec{w})$$

Unfortunately, maximizer is not characterized by a system of linear equations 😞
Numerical optimization

Fortunately, can find $\vec{w}$ with near-optimal log-likelihood using numerical optimization methods.
Numerical optimization

Fortunately, can find $\vec{w}$ with near-optimal log-likelihood using numerical optimization methods

- Objective function $\ln L(\vec{w})$ has “nice” properties: smoothness and concavity
Numerical optimization

Fortunately, can find $\vec{w}$ with near-optimal log-likelihood using numerical optimization methods

- Objective function $\ln L(\vec{w})$ has “nice” properties: smoothness and concavity
- “Slope” of objective at $\vec{w}$ is informative about how to improve $\vec{w}$ to achieve higher objective value
Fortunately, can find $\vec{w}$ with near-optimal log-likelihood using numerical optimization methods

- Objective function $\ln L(\vec{w})$ has “nice” properties: smoothness and concavity
- “Slope” of objective at $\vec{w}$ is informative about how to improve $\vec{w}$ to achieve higher objective value
- Repeated modifications to any initial $\vec{w}$ based on slope leads to near-optimal log-likelihood!
Example #1: Logistic regression (approx.) MLE for iris data
Example #1: Logistic regression (approx.) MLE for iris data

iteration $10^3$

- Ratio of sepal width to sepal length
- Ratio of petal width to petal length
- Setosa (0)
- Versicolor/Virginica (1)
- Decision boundary

Data points are plotted on a scatter plot with the ratio of sepal width to sepal length on the x-axis and the ratio of petal width to petal length on the y-axis. The decision boundary is represented by a dashed line.
Example #1: Logistic regression (approx.) MLE for iris data

iteration $10^4$

- Ratio of sepal width to sepal length
- Ratio of petal width to petal length
- Setosa (0)
- Versicolor/Virginica (1)
- Decision boundary

Graph showing a scatter plot of iris data with a decision boundary.
Example #1: Logistic regression (approx.) MLE for iris data

![Graph showing logistic regression for iris data with decision boundary and ratio of sepal width to sepal length and ratio of petal width to petal length.]

- Setosa (0)
- Versicolor/Virginica (1)

Iteration $10^5$
Example #1: Logistic regression (approx.) MLE for iris data

![Logistic regression graph](image)

- **Setosa (0)**
- **Versicolor/Virginica (1)**

Decision boundary:

- Ratio of sepal width to sepal length: 0.3, 0.35, 0.4, 0.45, 0.5, 0.55, 0.6, 0.65, 0.7, 0.75, 0.8
- Ratio of petal width to petal length: 0.05, 0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4, 0.45, 0.5

10^6 iterations.
Example #1: Logistic regression (approx.) MLE for iris data

- **Data Points**:
  - Ratio of sepal width to sepal length:
    - 0.3 0.35 0.4 0.45 0.5 0.55 0.6 0.65 0.7 0.75 0.8
  - Ratio of petal width to petal length:
    - 0.05 0.1 0.15 0.2 0.25 0.3 0.35 0.4 0.45 0.5

- **Class Labels**:
  - Setosa (0)
  - Versicolor/Virginica (1)

- **Decision Boundary**

---

**Graph Details**:
- **Axes**:
  - X-axis: Ratio of sepal width to sepal length
  - Y-axis: Ratio of petal width to petal length
- **Classes**:
  - Setosa (0) represented by blue squares
  - Versicolor/Virginica (1) represented by magenta crosses
- **Decision Boundary** represented by a black dashed line
Example #2: Text classification

- Two classes of articles posted to internet message boards: politics (1) and religion (0)
- Features:
  - Consider a vocabulary of $d = 34250$ words
  - Represent article by vector $\vec{x} \in \{0, 1\}^d$ where
    \[
    x_j = \mathbb{1}\{\text{article contains } j\text{-th vocabulary word}\}
    \]
- # training data: 3028; # test data: 2017

Log odds ratio (with $\vec{w} \in \mathbb{R}^d$):

\[ d \sum_{j} x_j w_j \]

Add up $w_j$'s for the set of (unique) words that appear in article.

- Words with positive $w_j > 0$ move the log odds ratio to favor politics (class 1)
- Words with negative $w_j < 0$ move the log odds ratio to favor religion (class 0)
Example #2: Text classification

- Two classes of articles posted to internet message boards: politics (1) and religion (0)
- Features:
  - Consider a vocabulary of $d = 34250$ words
  - Represent article by vector $\vec{x} \in \{0, 1\}^d$ where
    \[ x_j = 1\{\text{article contains } j\text{-th vocabulary word}\} \]
- # training data: 3028; # test data: 2017
- Log odds ratio (with $\vec{w} \in \mathbb{R}^d$):
  \[ \sum_{j=1}^{d} w_j x_j \]
  Add up $w_j$'s for the set of (unique) words that appear in article
Example #2: Text classification

- Two classes of articles posted to internet message boards: politics (1) and religion (0)
- Features:
  - Consider a vocabulary of $d = 34250$ words
  - Represent article by vector $\vec{x} \in \{0, 1\}^d$ where
    \[
    x_j = 1\{\text{article contains } j\text{-th vocabulary word}\}
    \]
- # training data: 3028; # test data: 2017
- Log odds ratio (with $\vec{w} \in \mathbb{R}^d$):
  \[
  \sum_{j=1}^{d} w_j x_j
  \]
  Add up $w_j$’s for the set of (unique) words that appear in article
  - Words with positive $w_j > 0$ move the log odds ratio to favor politics (class 1)
  - Words with negative $w_j < 0$ move the log odds ratio to favor religion (class 0)
Example #2: Logistic regression (approx.) MLE for text data

Using iterative numerical optimization algorithm to \textit{approximately} maximize log-likelihood

Training error rate of $f_{\hat{w}_{\text{mle}}}$: 0.9%
Test error rate of $f_{\hat{w}_{\text{mle}}}$: 8.5%
Example #2: Examining the learned weights

**Recall:** In logistic regression model with parameter $\vec{w}$,

- Words with positive $w_j > 0$ move the log odds ratio to favor politics class (1)
- Words with negative $w_j < 0$ move the log odds ratio to favor religion class (0)
Example #2: Examining the learned weights

Recall: In logistic regression model with parameter $\vec{w}$,

- Words with positive $w_j > 0$ move the log odds ratio to favor politics class (1)
- Words with negative $w_j < 0$ move the log odds ratio to favor religion class (0)

Examining the parameter vector $\vec{w}_{mle}$:

<table>
<thead>
<tr>
<th>Most positive coefficients</th>
<th>Most negative coefficients</th>
</tr>
</thead>
<tbody>
<tr>
<td>israel (+0.6019)</td>
<td>god (-1.1291)</td>
</tr>
<tr>
<td>gun (+0.5766)</td>
<td>christian (-0.7304)</td>
</tr>
<tr>
<td>government (+0.5620)</td>
<td>bible (-0.6747)</td>
</tr>
<tr>
<td>american (+0.5148)</td>
<td>jesus (-0.6679)</td>
</tr>
<tr>
<td>news (+0.4594)</td>
<td>keith (-0.5765)</td>
</tr>
<tr>
<td>clinton (+0.4417)</td>
<td>christians (-0.5295)</td>
</tr>
<tr>
<td>rights (+0.4178)</td>
<td>religion (-0.5285)</td>
</tr>
<tr>
<td>guns (+0.4169)</td>
<td>church (-0.4869)</td>
</tr>
<tr>
<td>israeli (+0.4166)</td>
<td>christ (-0.4635)</td>
</tr>
<tr>
<td>politics (+0.3933)</td>
<td>athos (-0.4456)</td>
</tr>
</tbody>
</table>
Example #2: Predictions of conditional probabilities

Recall: In logistic regression model with parameter \( \vec{w} \),

\[
\Pr_{\vec{w}}(Y = 1 \mid \vec{X} = \vec{x}) = \text{logistic}(\vec{x} \cdot \vec{w})
\]
Example of article with $\text{logistic}(\vec{x} \cdot \vec{w}_{\text{mle}}) \approx 0$:

Rick, I think we can safely say, 1) Robert is not the only person who understands the Bible, and 2), the leadership of the LDS church historically never has. Let’s consider some “personal interpretations” and see how much trust we should put in “Orthodox Mormonism”, which could never be confused with Orthodox Christianity. [...]
Example #2: Articles by predicted conditional probabilities

Example of article with $\text{logistic}(\vec{x} \cdot \vec{w}_{\text{mle}}) \approx 0$:

Rick, I think we can safely say, 1) Robert is not the only person who understands the Bible, and 2), the leadership of the LDS church historically never has. Let’s consider some “personal interpretations” and see how much trust we should put in “Orthodox Mormonism”, which could never be confused with Orthodox Christianity. […]

Example of article with $\text{logistic}(\vec{x} \cdot \vec{w}_{\text{mle}}) \approx 0.5$:

Does anyone know where I can access an online copy of the proposed “jobs” or “stimulus” legislation? Please E-mail me directly and if anyone else is interested, I can post this information.
Example of article with \( \text{logistic}(\vec{x} \cdot \vec{w}_{mle}) \approx 0 \):

Rick, I think we can safely say, 1) Robert is not the only person who understands the Bible, and 2), the leadership of the LDS church historically never has. Let’s consider some “personal interpretations” and see how much trust we should put in “Orthodox Mormonism”, which could never be confused with Orthodox Christianity. […]

Example of article with \( \text{logistic}(\vec{x} \cdot \vec{w}_{mle}) \approx 0.5 \):

Does anyone know where I can access an online copy of the proposed “jobs” or “stimulus” legislation? Please E-mail me directly and if anyone else is interested, I can post this information.

Example of article with \( \text{logistic}(\vec{x} \cdot \vec{w}_{mle}) \approx 1 \):

THE ENEMY WITHIN
~~~~~~~~~~~~~~~~
By Robert I. Friedman

| How The Anti-Defamation League Turned the Notion |
| of Human Rights on Its Head, Spying on Progress- |
| ives and Funneling Information to Law Enforcement |
Recap

- Logistic regression model is a natural linear model for binary classification data
  - Assumes log odds ratio are linear function of $x$.
- MLE in logistic regression can be efficiently approximated using numerical optimization methods.
Linear Classifiers
Geometry of linear classifiers

Decision boundary of linear classifier $f_{\bar{w}}(\bar{x}) = \mathbb{1}\{\bar{x} \cdot \bar{w} > 0\}$ is a **hyperplane** in $\mathbb{R}^d$

$$H := \{\bar{x} \in \mathbb{R}^d : \bar{x} \cdot \bar{w} = 0\}$$

(assuming $\bar{w} \neq \bar{0}$)

- Hyperplane $H$ is a $(d - 1)$-dimensional (linear) subspace of $\mathbb{R}^d$
  - $H$ passes through $\bar{0}$
- $\bar{w}$ is **normal** vector for $H$ (and provides an orientation)
- Angle $\alpha$ between $\bar{x}$ and $\bar{w}$ satisfies

$$\cos(\alpha) = \frac{\bar{x} \cdot \bar{w}}{\|\bar{x}\|_2 \|\bar{w}\|_2}$$
Feature expansion

With feature expansions, can get different types of decision boundaries
(viewed in terms of features prior to expansion)
Feature expansion

With feature expansions, can get different types of decision boundaries
(viewed in terms of features prior to expansion)

Examples:

- **Affine expansion**: Hyperplane boundary that need not pass through $\vec{0}$
Feature expansion

With feature expansions, can get different types of decision boundaries (viewed in terms of features prior to expansion)

Examples:

- **Affine expansion**: Hyperplane boundary that need not pass through $\vec{0}$

- **Quadratic expansion (in $\mathbb{R}^2$)**: Boundary can be any conic section (ellipse, hyperbola, parabola)
  \[
  \varphi(x_1, x_2) = (x_1^2, x_2^2, x_1 x_2, x_1, x_2, 1) \in \mathbb{R}^6
  \]
Relationship between models for binary classification:
Empirical risk minimization for linear classification:

Given training data $S$, find linear classifier $f_{\vec{w}}(\vec{x}) = 1\{\vec{x} \cdot \vec{w} > 0\}$ to minimize training error rate

$$\frac{\# \text{ examples } (\vec{x}, y) \in S \text{ such that } f_{\vec{w}}(\vec{x}) \neq y}{|S|}$$
Empirical risk minimization for linear classification:

Given training data \( S \), find linear classifier \( f_{\vec{w}}(\vec{x}) = 1\{\vec{x} \cdot \vec{w} > 0\} \) to minimize training error rate

\[
\frac{\# \text{ examples } (\vec{x}, y) \in S \text{ such that } f_{\vec{w}}(\vec{x}) \neq y}{|S|}
\]

**Theorem.** Under IID model for training data, ERM linear classifier \( f_{\vec{w}_{\text{erm}}} \) “typically” satisfies

\[
[\text{True error rate of } f_{\vec{w}_{\text{erm}}}] \leq \min_{\vec{w} \in \mathbb{R}^d} [\text{True error rate of } f_{\vec{w}}] + [\text{a small number if } n \gg d]
\]
Recall: For linear regression, OLS = MLE in normal linear regression model = ERM with squared error
Recall: For linear regression, OLS = MLE in normal linear regression model = ERM with squared error

Question (regarding linear classification):

Does logistic regression MLE give the linear classifier of minimum training error rate?
Recall: For linear regression, OLS = MLE in normal linear regression model = ERM with squared error

Question (regarding linear classification):

Does logistic regression MLE give the linear classifier of minimum training error rate?

Answer:

▶ No, in general 😊
Recall: For linear regression, OLS = MLE in normal linear regression model = ERM with squared error

Question (regarding linear classification):

*Does logistic regression MLE give the linear classifier of minimum training error rate?*

Answer:

▶ No, in general 😞

▶ For any efficient algorithm, there's some data set for which the algorithm won't be able to find the ERM linear classifier
Recall: For linear regression, OLS = MLE in normal linear regression model = ERM with squared error.

Question (regarding linear classification):

Does logistic regression MLE give the linear classifier of minimum training error rate?

Answer:

- No, in general 😞
  - For any efficient algorithm, there's some data set for which the algorithm won't be able to find the ERM linear classifier.
- Yes, if the training data is linearly separable: i.e., the data set is one for which the ERM linear classifier has training error rate zero (called a linear separator).
Linear separability

**Linearly separable** with affine feature expansion

Setosa (0)
Versicolor/Virginica (1)
decision boundary
Linear separability

Not linearly separable with affine feature expansion

Class 0
Class 1
decision boundary
Linear separability

Linearly separable with quadratic feature expansion
Perceptron

Algorithm for learning a linear classifier on data \((\vec{x}_1, y_1), \ldots, (\vec{x}_n, y_n)\) that is linearly separable

Frank Rosenblatt, 1958
Perceptron

Algorithm for learning a linear classifier on data \((\vec{x}_1, y_1), \ldots, (\vec{x}_n, y_n)\) that is linearly separable

**Perceptron**

- Initialize \(\vec{w} := \vec{0} \in \mathbb{R}^d\)
- Loop:
  - Pick any example \((\vec{x}_i, y_i)\) such that \(f_{\vec{w}}(\vec{x}_i) \neq y_i\) (i.e., an example misclassified by \(f_{\vec{w}}\))
  - (If there is no such example, halt and return \(\vec{w}\))
  - Update \(\vec{w}\):
    \[
    \vec{w} := \begin{cases} 
    \vec{w} + \vec{x}_i & \text{if } y_i = 1 \\
    \vec{w} - \vec{x}_i & \text{if } y_i = 0
    \end{cases}
    \]
Perceptron

Algorithm for learning a linear classifier on data \((\vec{x}_1, y_1), \ldots, (\vec{x}_n, y_n)\) that is linearly separable

**Perceptron**

- Initialize \(\vec{w} := \vec{0} \in \mathbb{R}^d\)
- Loop:
  - Pick any example \((\vec{x}_i, y_i)\) such that \(f_{\vec{w}}(\vec{x}_i) \neq y_i\) (i.e., an example misclassified by \(f_{\vec{w}}\))
  - (If there is no such example, halt and return \(\vec{w}\))
  - Update \(\vec{w}\):
    \[
    \vec{w} := \begin{cases} 
    \vec{w} + \vec{x}_i & \text{if } y_i = 1 \\
    \vec{w} - \vec{x}_i & \text{if } y_i = 0
    \end{cases}
    \]

**Caution:** Will loop forever if data is not linearly separable!
Example #1: Iris classification, revisited

**Perceptron** finds affine classifier with zero classification mistakes on training data.
Example #1: Iris classification, revisited

- **Perceptron** finds affine classifier with zero classification mistakes on training data.
- **Number of iterations:** 403
Example #1: Iris classification, revisited

- **Perceptron** finds affine classifier with zero classification mistakes on training data
- Number of iterations: 403
- Number of mistakes does not decrease monotonically, but does eventually reach 0
Example #2: Text classification, revisited

- Two classes of articles posted to internet message boards: politics (1) and religion (0)
- Number of features: $d = 34250$
- Linear classifier $f_w$ found via Perceptron using $n = 3028$ training examples after 757 iterations
Example #2: Text classification, revisited

- Two classes of articles posted to internet message boards: politics (1) and religion (0)
- Number of features: $d = 34250$
- Linear classifier $f_{\vec{w}}$ found via Perceptron using $n = 3028$ training examples after 757 iterations
Example #2: Text classification, revisited

- Two classes of articles posted to internet message boards: politics (1) and religion (0)
- Number of features: $d = 34250$
- Linear classifier $f_{\mathbf{w}}$ found via Perceptron using $n = 3028$ training examples after 757 iterations

Training error rate of $f_{\mathbf{w}}$: 0%
Test error rate of $f_{\mathbf{w}}$: 9.4%
Example #2: Text classification, revisited

- Two classes of articles posted to internet message boards: politics (1) and religion (0)
- Number of features: \( d = 34250 \)
- Linear classifier \( f_{\vec{w}} \) found via Perceptron using \( n = 3028 \) training examples after 757 iterations

**Observations:**

- Training data is linearly separable
  - I didn’t know this *a priori*!
  - (Note: If I had been more patient with the optimization method for logistic regression MLE, I would have discovered this.)
Example #2: Text classification, revisited

- Two classes of articles posted to internet message boards: politics (1) and religion (0)
- Number of features: $d = 34250$
- Linear classifier $f_w$ found via Perceptron using $n = 3028$ training examples after 757 iterations

Observations:
- Training data is linearly separable
  - I didn’t know this \textit{a priori}!
  - (Note: If I had been more patient with the optimization method for logistic regression MLE, I would have discovered this.)
- Test error rate (9.4\%) is reasonable even though $n \ll d$ (!)
  - Still, it is much higher than training error rate (0\%)
Recap

- Linear classifiers separate the feature space using a hyperplane boundary.
- Although statistical learning concepts extend to linear classification, ERM for linear classifiers is very different from MLE for logistic regression.
  - ERM for linear classifiers is tractable in special case where data is linearly separable.
  - Perceptron can find a linear separator if one exists.
    - There are many variants of Perceptron! We’ll see some in next homework assignment.
    - Perceptron is also related to “Stochastic Gradient Method”, which we’ll see later in the course.