THE UNIVERSITY OF TOKYO

数理情報工学特論第一
【機械学習とデータマイニング】
3章：分類②

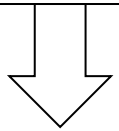
かしま　ひさし
鹿島 久嗣
（数理 6 研）
kashima@mist.i.～

東京大学
THE UNIVERSITY OF TOKYO

DEPARTMENT OF MATHEMATICAL INFORMATICS

「分類」について学びます

- 分類問題の定義
- 分類問題の応用
- 分類のためのモデル：ロジスティック回帰
- 分類問題の定式化
- 学習アルゴリズム：ニュートン法と最急勾配法



- パーセプトロン
- マージン最大化学習
- 多クラス分類
- 勾配法による多クラスロジスティック回帰の学習
- 多クラスパーセプトロン

パーセプトロン

最尤推定よりもゆるい目標：正解に負正解よりも高い確率を与える

- 最尤推定：モデルが訓練データに与える確率（の対数）を最大化するようにパラメータを決定する

$$\mathbf{w}^* = \arg\max_{\mathbf{w}} \frac{1}{N} \sum_{i=1}^{N} \log P(y^{(i)} | \boldsymbol{\phi}(x^{(i)}); \mathbf{w})$$

  - 最尤推定は、入力 $x^{(i)}$ に対して、正しい出力 $y = y^{(i)}$ にモデルが与える確率を最大化することを目指している

- もう少しルーズな目標を考える：正しい出力（$y=y^{(i)}$）に対して与える確率を正しくない出力（$y \neq y^{(i)}$）に対して与える確率より高くする

$$\log P(y^{(i)} | \boldsymbol{\phi}(x^{(i)}); \mathbf{w}) > \log P(y | \boldsymbol{\phi}(x^{(i)}); \mathbf{w}) \text{ for } y \neq y^{(i)}$$

  - 条件付き確率に基づいて y を +1 か -1のどちらかに予測する場合、上の大小関係が成り立ちさえすれば、予測は間違えない

- 最尤推定は、予測が当たるだけに飽き足らず、一層の高みを目指しているともいえる

2クラスロジスティック回帰の場合、先の（ゆるい）目標はパラメータで張られる超平面での分割を求めることに一致します

- $P(y^{(N+1)}|\phi(x^{(N+1)}); \mathbf{w})$ が2クラスのロジスティック回帰であるとする
- 正解のクラスが $y^{(i)}$=1である場合、$y^{(i)} = 1$ である確率が $y^{(i)} = -1$である確率よりも大きいという条件は：

$$\log \frac{1}{1 + \exp(-\mathbf{w}^\top \phi(x^{(i)}))} > \log \frac{\exp(-\mathbf{w}^\top \phi(x^{(i)})}{1 + \exp(-\mathbf{w}^\top \phi(x^{(i)}))}$$

- これを整理すると： $\mathbf{w}^\top \phi(x^{(i)}) > 0$
- 一方、正解が $y^{(i)} = -1$の場合は： $\mathbf{w}^\top \phi(x^{(i)}) < 0$
- つまり、上の制約は、超平面 $\mathbf{w}^\top \phi(x^{(i)}) = 0$を境に正例と負例を正しく分割することを要求していることになる
- 2つの制約をまとめて書くと： $y^{(i)}\ \mathbf{w}^\top \phi(x^{(i)}) > 0$

パーセプトロンは「正解に負正解よりも高い確率を与える」を実現するシンプルな逐次学習型アルゴリズムです

- パーセプトロンは 制約 $y^{(i)}\ \mathbf{w}^{\top}\boldsymbol{\phi}(x^{(i)}) > 0$ を実現するアルゴリズム
- 訓練データを1つづつ処理する**逐次学習**型のアルゴリズム
- 学習の各ステップにおいて、1つの訓練データ($x^{(i)}, y^{(i)}$)を選び、これを用いてモデルパラメータ $\mathbf{w}$ の更新を行う
- ($x^{(i)}, y^{(i)}$) について既に制約が満たされている場合には何も行わない
- 制約が満たされない場合に限って以下の更新式を用いてパラメータの更新を行う

$$\mathbf{w}^{\mathrm{NEW}} \leftarrow \mathbf{w}^{\mathrm{OLD}} + y^{(i)}\boldsymbol{\phi}(x^{(i)})$$

  - パラメータの更新は制約を満たす方向に行われる（後述）

パーセプトロンのアルゴリズム

アルゴリズム 1：2 クラス分類におけるパーセプトロンアルゴリズム

1: パラメータの初期化 $\boldsymbol{w} = \boldsymbol{0}$
2: ランダムに i 番目の訓練データ $(x^{(i)}, y^{(i)})$ を選ぶ
3: もしも制約

$$y^{(i)} \boldsymbol{w}^{\top} \boldsymbol{\phi}(x^{(i)}) > 0$$

が満たされているならば、何もせずにステップ 2 へ戻る
4: 制約が満たされないならば、パラメータ更新式

$$w^{\mathrm{NEW}} \leftarrow w^{\mathrm{OLD}} + y^{(i)} \boldsymbol{\phi}(x^{(i)})$$

によってパラメータを更新し、ステップ 2 に戻る

パーセプトロンのパラメータ更新則は「正解の確率＞負正解の確率」の制約を満たそうとする方向に行われます

- 更新後のパラメータを用いて、入力 $x^{(i)}$ に対する予測 $\mathbf{w}^{\mathrm{NEW}\top}\boldsymbol{\phi}(x^{(i)})$ を計算してみると：

$$\mathbf{w}^{\mathrm{NEW}\top}\boldsymbol{\phi}(x^{(i)}) = \mathbf{w}^{\mathrm{OLD}\top}\boldsymbol{\phi}(x^{(i)}) + y^{(i)}\boldsymbol{\phi}(x^{(i)})^{\top}\boldsymbol{\phi}(x^{(i)})$$

- 正しいクラスが $y^{(i)} = +1$ の場合には $\mathbf{w}^{\mathrm{NEW}\top}\boldsymbol{\phi}(x^{(i)})$ が $\mathbf{w}^{\mathrm{OLD}\top}\boldsymbol{\phi}(x^{(i)})$ よりも $\boldsymbol{\phi}(x^{(i)})^{\top}\boldsymbol{\phi}(x^{(i)})$ (必ず0以上の値をとる) だけ大きくなる
- 一方、正しいクラスが $y^{(i)} = -1$ の場合には同じ量だけ小さくなる
- つまり、制約を満たそうとする方向にパラメータが更新されている

パーセプトロンは、制約を満たすパラメータを有限回のパラメータ更新で見つけることができます（もしあれば）

- パーセプトロンアルゴリズムのパラメータ更新回数についての定理
- 2つの条件が満たされていると仮定する：

  [条件1] 全てのデータに対し制約 $y^{(i)}\ \mathbf{w}^{*\top}\phi(x^{(i)}) > 0$を満たす、ある理想的なパラメータ$\mathbf{w}^*$が存在して：

  - ある正の定数 $\gamma > 0$ について $y^{(i)}\ \mathbf{w}^{*\top}\phi(x^{(i)}) > \gamma$ が成立する
  - $\mathbf{w}^*$ の2-ノルムは$\|\mathbf{w}^*\|_2^2 = 1$である

  [条件2] 全ての訓練データについて特徴ベクトル $\phi(x^{(i)})$ のノルムが $\|\phi(x^{(i)})\|_2^2 < R^2$ である

- このとき、パーセプトロンアルゴリズムが全ての制約を満たすパラメータを発見するまでのパラメータ更新回数 k は高々：$\left(\frac{R}{\gamma}\right)^2$

  （予測の符号を間違える回数）

証明（前半）

- 今 k 回目のパラメータ更新（k 回目の制約不成立）が起こったとする
- 更新後のパラメータ$\mathbf{w}^{\mathrm{NEW}}$と、理想的なパラメータ$\mathbf{w}^*$との近さを見るために、その内積 $\mathbf{w}^{\mathrm{NEW}\top}\mathbf{w}^*$を調べると：

$$\begin{aligned}
\mathbf{w}^{\mathrm{NEW}\top}\mathbf{w}^* &= \mathbf{w}^{\mathrm{OLD}\top}\mathbf{w}^* + y^{(i)}\boldsymbol{\phi}(x^{(i)})^\top\mathbf{w}^* && \text{：更新式の定義より} \\
&> \mathbf{w}^{\mathrm{OLD}\top}\mathbf{w}^* + \gamma && \text{：仮定 } y^{(i)}\,\mathbf{w}^\top\phi(x^{(i)}) > \gamma \text{ より} \\
&> k\gamma && \text{：パラメータ更新回数が } k \text{ 回であることより}
\end{aligned}$$

- パラメータ更新のたびに内積が少なくとも γ づつ大きくなっていく
- 両辺を$\|\mathbf{w}^{\mathrm{NEW}}\|_2$で割ると：$\left(\frac{\mathbf{w}^{\mathrm{NEW}}}{\|\mathbf{w}^{\mathrm{NEW}}\|_2}\right)^\top\mathbf{w}^* > \frac{k\gamma}{\|\mathbf{w}^{\mathrm{NEW}}\|_2}$
  - 左辺にある2つのベクトルの 2-ノルムは共に1
  - その内積は1以下であるから $\frac{k\gamma}{\|\mathbf{w}^{\mathrm{NEW}}\|_2} \leq 1$

証明（後半）

- あとは$\|\mathbf{w}^{\mathrm{NEW}}\|_2$を評価すればよい。
- 更新式 $\mathbf{w}^{\mathrm{NEW}} = \mathbf{w}^{\mathrm{OLD}} + y^{(i)}\boldsymbol{\phi}(x^{(i)})$ より：

$$\begin{aligned}
\|\mathbf{w}^{\mathrm{NEW}}\|_2^2 &= \|\mathbf{w}^{\mathrm{OLD}}\|_2^2 + \|\boldsymbol{\phi}(x^{(i)})\|_2^2 + 2y^{(i)}\mathbf{w}^{\mathrm{OLD}\top}\boldsymbol{\phi}(x^{(i)}) \\
&\leq \|\mathbf{w}^{\mathrm{OLD}}\|_2^2 + R^2 + 2y^{(i)}\mathbf{w}^{\mathrm{OLD}\top}\boldsymbol{\phi}(x^{(i)}) \quad : \text{条件2 } \|\phi(x^{(i)})\|_2{}^2 < R^2 \\
&\leq \|\mathbf{w}^{\mathrm{OLD}}\|_2^2 + R^2 \quad : \text{制約の不成立の仮定より}
\end{aligned}$$

- パラメータの更新回数が k 回であることから、この不等式を繰り返し適用することにより $\|\mathbf{w}^{\mathrm{NEW}}\|_2{}^2 \leq kR^2$ すなわち $\|\mathbf{w}^{\mathrm{NEW}}\|_2 \leq \sqrt{k}\,R$
- これを、前頁で導いた不等式と組み合わせ、γ について解くと：

$$\frac{k\gamma}{\|\mathbf{w}^{\mathrm{NEW}}\|_2} \leq 1 \qquad k \leq \left(\frac{R}{\gamma}\right)^2$$

データを完璧に分類できる超平面が存在しない場合でも、パーセプトロンの性能はよいことが知られています

- 定理では、制約 $y^{(i)}\ \mathbf{w}^{*\top}\phi(x^{(i)}) > 0$が成立すること、すなわち、全ての訓練データを間違いなく判別することのできる超平面 $\mathbf{w}^{*\top}\phi(x^{(i)}) > 0$ が存在することを仮定していた
- 現実的には真実の $P(y|x)$ は線形モデルよりももっと複雑であったり、あるいはデータにノイズが含まれていたりなどの事情により、この制約は必ず満たされるとは限らない
- このような場合にも、「最も制約 を破らないようなパラメータ$\mathbf{w}^*$」と比較して、パーセプトロンアルゴリズムが誤った予測をする回数が小さく抑えられることがわかっている

$$k \leq \left(\frac{R+D}{\gamma}\right)^2$$

  - ここでDは $\mathbf{w}^*$ の制約破りっぷりを表す

マージン最大化

パーセプトロンは「マージン」を正にするように働きます

- パーセプトロンが満たそうとする制約：

$$\log P(y^{(i)}|\phi(x^{(i)});\mathbf{w}) > \log P(y|\phi(x^{(i)});\mathbf{w}) \text{ for } y \neq y^{(i)}$$

$$\Downarrow$$

$$\log P(y^{(i)}|x^{(i)};\mathbf{w}) - \log P(y|x^{(i)};\mathbf{w}) > 0 \text{ for } y \neq y^{(i)}$$

は、全ての訓練データ$\{(x^{(i)}, y^{(i)})\}_{i=1}^{N}$に対し
「正解に対する確率と不正解に対する確率の差が0より大」
が、成り立つことと見ることができる

- つまり：

$$\Delta(\mathbf{w}) \equiv \min_{i\in\{1,2,\ldots,N\}} \min_{y\neq y^{(i)}} \log P(y^{(i)}|x^{(i)};\mathbf{w}) - \log P(y|x^{(i)};\mathbf{w})$$

としたときに$\Delta(\mathbf{w}) > 0$が成り立つということ

  - この$\Delta(\mathbf{w})$は**マージン**と呼ばれる
  - マージンは**w**によって変化する

マージンを最大化すると良いような気がしますが、そのままでは定式化に問題があります

- パーセプトロン（や最尤推定の）とは別の学習目標として、マージン$\Delta(\mathbf{w})$を「最大化」するということが考えられる。
  - パーセプトロンの掲げる目標である、マージンが0より大きいということだけに飽き足らず、これをさらに推し進めることを目指す
- マージン最大化を最適化問題として書けば以下のように書けそう：

$$\mathbf{w}^* = \underset{w}{\operatorname{argmax}} \ \Delta(\mathbf{w})$$

- 2クラスのロジスティック回帰モデルを用いる場合のマージンは

$$\Delta(\mathbf{w}) \equiv \min_{i \in \{1,2,\ldots,N\}} y^{(i)} \mathbf{w}^\top \boldsymbol{\phi}(x^{(i)})$$

- しかし、ここで1つ問題が起こる：
  - $\Delta(\mathbf{w}) > 0$を満たす$\mathbf{w}$が存在したとすると、$\mathbf{w}$の方向を変えずに大きくすることで、マージンをいくらでも大きくできてしまう

マージンを固定してパラメータのノルムを小さくすることでマージン最大化を2次計画問題として定式化できます

- 少し考え方を変えてみる
- $\mathbf{w}$のノルムを大きくすることでマージンをいくらでも大きくできてしまうのなら、逆にマージン$\Delta(\mathbf{w})$を（例えば1に）固定してしまい$\mathbf{w}$のノルムをできる限り小さくすることにする：

$$\mathbf{w}^* = \arg\min_{w} \ \| \mathbf{w} \|_2^2$$
$$\text{s.t.} \ \ \Delta(\mathbf{w}) = 1$$

  - $\Delta(\mathbf{w}) = 1$としたのには深い意味はない（2でも3でも何でもよい）
- マージンの定義から、制約式の方を少し書き換えると：

$$\mathbf{w}^* = \arg\min_{w} \ \| \mathbf{w} \|_2^2$$
$$\text{s.t.} \ \ y^{(i)} \mathbf{w}^\top \phi(x^{(i)}) \geq 1 \ \ (\text{for } i = 1, 2, \ldots, N)$$

- この最適化問題は、**2次計画問題**と呼ばれる最適化問題

データを完璧に分類できる超平面が存在しない場合には「制約がなるべく成り立つように」定式化します

- マージン最大化の定式化は、$\Delta(\mathbf{w}^*) > 0$ が成り立つ$\mathbf{w}^*$が存在するときには良さそうだが、この制約は必ずしも満たされるとは限らない
- この制約を緩め「制約がなるべく成り立つように」要請する
- 制約は$y^{(i)}\ \mathbf{w}^\top \phi(x^{(i)}) \geq 1$を要求しているため、これが破られたときのペナルティの大きさを$1 - y^{(i)}\ \mathbf{w}^\top \phi(x^{(i)})$によって測ることにする
- 制約が満たされた時（ペナルティが0）と纏めて書くと、i番目の訓練データに対するペナルティの大きさは $\max\{1 - y^{(i)}\ \mathbf{w}^\top \phi(x^{(i)}), 0\}$
- ノルムも併せて最小化すると、最適化問題は：

$$\mathbf{w}^* = \underset{w}{\operatorname{argmin}} \frac{1}{N} \sum_{i=1}^{N} \max\{1 - y^{(i)} \mathbf{w}^\top \phi(x^{(i)}), 0\} + \lambda \parallel \mathbf{w} \parallel_2^2$$

  - この形は事後確率最大化のときの対数尤度の項と正則化項に対応

多クラス分類

多クラスのロジスティック回帰モデル：クラスの数だけパラメータベクトルを用意します

- ロジスティック回帰モデルを、多クラスの場合に拡張する
  - 後に配列データのラベル付けモデル（条件付確率場）に拡張される
- クラスの集合 $\mathcal{Y}$ を$\{1, 2, \ldots, C\}$に対するCクラスのロジスティック回帰モデルは：

$$P(y|x; \{\mathbf{w}^{(c)}\}_{c\in\mathcal{Y}}) \equiv \frac{\exp(\mathbf{w}^{(y)\top}\boldsymbol{\phi}(x))}{\sum_{c\in\mathcal{Y}}\exp(\mathbf{w}^{(c)\top}\boldsymbol{\phi}(x))}$$

  - 各クラス $c \in \mathcal{Y}$ ごとにD次元のパラメータベクトル $\mathbf{w}^{(c)}$が定義されモデル全体としては C D個のパラメータがあることになる
  - expの中には線形回帰モデルの形が現れており、$\mathbf{w}^{(c)}$ は特徴ベクトルのそれぞれの特徴のクラス c への貢献度を表す
  - 指数を取ることで0以上の値に変換し、さらに正規化することで (0,1) の間の確率値に変換している

多クラスロジスティック回帰モデルは、特徴空間を複数の超平面で切り分けます

- 多クラスロジスティック回帰を用いてクラス予測を行うときには：

$$y^* \equiv \underset{y \in \mathcal{Y}}{\text{argmax}} \; P(y|x; \{\mathbf{w}^{(c)}\}_{c \in \mathcal{Y}})$$

- つまり $P(y^*|x; \{\mathbf{w}^{(c)}\}_{c \in \mathcal{Y}}) > P(y|x; \{\mathbf{w}^{(c)}\}_{c \in \mathcal{Y}})$ すなわち $(\mathbf{w}^{(y^*)} - \mathbf{w}^{(y)})^\top \boldsymbol{\phi}(x) > 0$ がすべての $y \neq y^*$ について成立するとき y^* を出力する
- $(\mathbf{w}^{(y^*)} - \mathbf{w}^{(y)})^\top \boldsymbol{\phi}(x) > 0$ は超平面の片側を指すため、その集合は複数の超平面で切りだされた領域となる
- $\boldsymbol{\phi}(x^{(i)})$ が丁度超平面上にある場合以外には、不等式を満たす y^* は必ず存在するので、特徴空間が複数の超平面によって切り分けられていることになる
- 一方、2クラス分類に帰着する方法では、$P(y = +1|x; \mathbf{w}) = 0.5$ を境目とした判断方法では、どのクラスにも属さない場合がありうる

多クラスロジスティック回帰モデルには、等価な別の表記があります

- ロジスティック回帰モデルは以下のように書き直すことができる

$$P(y|x;\omega) \equiv \frac{\exp(\omega^\top \boldsymbol{\varphi}(x,y))}{\sum_{c\in\mathcal{Y}} \exp(\omega^\top \boldsymbol{\varphi}(x,c))}$$

  - パラメータωはDC次元のベクトル：$\omega \equiv (\mathbf{w}^{(1)\top}, \mathbf{w}^{(2)\top}, \ldots, \mathbf{w}^{(C)\top})^\top$
  - 特徴ベクトル $\boldsymbol{\varphi}(x,y)$ もDC次元のベクトル：

$$\boldsymbol{\varphi}(x,y) \equiv (\delta(y=1)\boldsymbol{\phi}(x)^\top, \delta(y=2)\boldsymbol{\phi}(x)^\top, \ldots, \delta(y=C)\boldsymbol{\phi}(x)^\top)^\top$$

    - x と y の組み合わせに対して定義される
    - $\delta(\cdot)$：括弧の中身が成立するなら1を、しないならば0をとる
- クラス毎のパラメータではなく、特徴ベクトルの側でクラスの違いを吸収することで、パラメータベクトルを1つにしている
  - 後に紹介する（クラスが実質いくらでもある）条件付き確率場の基本的な形式となる

勾配法による多クラスロジスティック回帰の学習

多クラスロジスティック回帰モデルを最急勾配法で学習するために勾配を計算しておきます

- 多クラスのロジスティック回帰の場合の目的関数は：

$$L(\{\mathbf{w}^{(c)}\}_{c\in\mathcal{Y}}) \equiv \frac{1}{N}\sum_{i=1}^{N} \log P(y^{(i)}|\boldsymbol{\phi}(x^{(i)}); \{\mathbf{w}^{(c)}\}_{c\in\mathcal{Y}}) - \lambda \sum_{c\in\mathcal{Y}} \| \mathbf{w}^{(c)} \|_2^2$$

$$= \frac{1}{N}\sum_{i=1}^{N} \log \frac{\exp(\mathbf{w}^{(y^{(i)})^\top}\boldsymbol{\phi}(x^{(i)}))}{\sum_{c\in\mathcal{Y}}\exp(\mathbf{w}^{(c)^\top}\boldsymbol{\phi}(x^{(i)}))} - \lambda \sum_{c\in\mathcal{Y}} \| \mathbf{w}^{(c)} \|_2^2$$

$$= \frac{1}{N}\sum_{i=1}^{N} \left(\mathbf{w}^{(y^{(i)})^\top}\boldsymbol{\phi}(x^{(i)}) - \log \sum_{c\in\mathcal{Y}} \exp(\mathbf{w}^{(c)^\top}\boldsymbol{\phi}(x^{(i)})) \right) - \lambda \sum_{c\in\mathcal{Y}} \| \mathbf{w}^{(c)} \|_2^2$$

- これをクラス y のパラメータ $\mathbf{w}^{(y)}$で偏微分してみると：

$$\frac{\partial L(\{\mathbf{w}^{(c)}\}_{c\in\mathcal{Y}})}{\partial \mathbf{w}^{(y)}} = \frac{1}{N}\sum_{i=1}^{N} \left(\delta(y = y^{(i)})\boldsymbol{\phi}(x^{(i)}) - \frac{\exp(\mathbf{w}^{(y)^\top}\boldsymbol{\phi}(x^{(i)}))}{\sum_{c\in\mathcal{Y}}\exp(\mathbf{w}^{(c)^\top}\boldsymbol{\phi}(x^{(i)}))}\boldsymbol{\phi}(x^{(i)}) \right) - 2\lambda\mathbf{w}^{(y)}$$

$$= \frac{1}{N}\sum_{i=1}^{N} \left(\delta(y = y^{(i)}) - P(y|\boldsymbol{\phi}(x^{(i)}); \{\mathbf{w}^{(c)}\}_{c\in\mathcal{Y}}) \right) \boldsymbol{\phi}(x^{(i)}) - 2\lambda\mathbf{w}^{(y)}$$

  - $\delta(\cdot)$：括弧の中身が成立するなら1を、しないならば0をとる

「(log) ∑ exp」の計算を行うのにはちょっとしたテクニックが必要です

- 多クラスロジスティック回帰による予測時の確率や学習時の収束判定などのために対数尤度の項を計算する際に現れる

$$\sum_{c \in \mathcal{Y}} \exp(\mathbf{w}^{(c)\top} \boldsymbol{\phi}(x)) \quad \text{もしくは} \quad \log \sum_{c \in \mathcal{Y}} \exp(\mathbf{w}^{(c)\top} \boldsymbol{\phi}(x))$$

では、expの中身が大きいときに計算の途中で桁あふれを起こしてしまう問題がある

- この問題を回避するために、以下の等式を用いることで指数の中身が大きくなることを防ぐことができる

$$\log(\exp(a) + \exp(b)) = \max\{a, b\} + \log\left(\exp\left(\min\{a, b\} - \max\{a, b\}\right) + 1\right)$$

多クラスのパーセプトロン

多クラスのパーセプトロンもまた、「正解に負正解よりも高い確率を与える」制約を逐次学習によって実現します

- パーセプトロンが満たそうとする制約：

$$\log P(y^{(i)}|\phi(x^{(i)});\mathbf{w}) > \log P(y|\phi(x^{(i)});\mathbf{w}) \text{ for } y \neq y^{(i)}$$

は多クラスロジスティック回帰の場合は、全ての $y \neq y^{(i)}$ について：

$$(\mathbf{w}^{(y^{(i)})} - \mathbf{w}^{(y)})^\top \phi(x^{(i)}) > 0$$

つまり

$$\mathbf{w}^{(y^{(i)})^\top} \phi(x^{(i)}) - \max_{y \in \mathcal{Y}} \mathbf{w}^{(y)^\top} \phi(x^{(i)}) > 0$$

- 従って、2クラスの場合と同じように、多クラスパーセプトロンは毎ステップにおいて、ひとつの訓練データ $(x^{(i)}, y^{(i)})$ に対し：
  - 上の制約が成り立つ場合には何もしない
  - 制約が成り立たない場合に限り、この制約を成り立たせる方向にパラメータの更新を行う

パラメータ更新は制約を成り立たせる方向に行います

- 具体的な更新則としては、予測値が最大となるクラス $\tilde{y}$ を：

$$\tilde{y} \equiv \max_{y \in \mathcal{Y}} \mathbf{w}^{(y)\top} \boldsymbol{\phi}(x^{(i)})$$

とすると、更新式は：

$$\mathbf{w}^{(y^{(i)})^{\mathrm{NEW}}} \leftarrow \mathbf{w}^{(y^{(i)})^{\mathrm{OLD}}} + \boldsymbol{\phi}(x^{(i)})$$

$$\mathbf{w}^{(\tilde{y})^{\mathrm{NEW}}} \leftarrow \mathbf{w}^{(\tilde{y})^{\mathrm{OLD}}} - \boldsymbol{\phi}(x^{(i)})$$

- この更新は結果とし制約を成り立たせる方向に向かう
  - $\mathbf{w}^{(y^{(i)})^{\mathrm{NEW}\top}} \boldsymbol{\phi}(x^{(i)})$ は $\mathbf{w}^{(y^{(i)})^{\mathrm{OLD}\top}} \boldsymbol{\phi}(x^{(i)})$ よりも大きくなるように
  - $\mathbf{w}^{(\tilde{y})^{\mathrm{NEW}\top}} \boldsymbol{\phi}(x^{(i)})$ は $\mathbf{w}^{(\tilde{y})^{\mathrm{OLD}\top}} \boldsymbol{\phi}(x^{(i)})$ よりも小さくなるように

多クラスパーセプトロンのアルゴリズム

アルゴリズム 2：多クラス分類におけるパーセプトロンアルゴリズム

1: パラメータの初期化 $\{\boldsymbol{w}^{(c)} = \mathbf{0}\}_{c=1}^{C}$

2: ランダムに i 番目の訓練データ $(x^{(i)}, y^{(i)})$ を選ぶ

3: 予測が最大になるクラス $\tilde{y} \equiv \underset{y}{\operatorname{argmax}}\ \boldsymbol{w}^{(y)\top}\boldsymbol{\phi}(x^{(i)})$ を見つける

4: もしも制約

$$(\boldsymbol{w}^{(y^{(i)})} - \boldsymbol{w}^{(\tilde{y})})^{\top}\boldsymbol{\phi}(x^{(i)}) > 0$$

が満たされているならば、何もせずにステップ 2 へ戻る

5: 制約が満たされないならば、パラメータ更新式

$$\boldsymbol{w}^{(y^{(i)})^{\mathrm{NEW}}} \leftarrow \boldsymbol{w}^{(y^{(i)})^{\mathrm{OLD}}} + \boldsymbol{\phi}(x^{(i)})$$
$$\boldsymbol{w}^{(\tilde{y})^{\mathrm{NEW}}} \leftarrow \boldsymbol{w}^{(\tilde{y})^{\mathrm{OLD}}} - \boldsymbol{\phi}(x^{(i)})$$

によってパラメータを更新し、ステップ 2 に戻る