

## E資格黒本 19章

1. 固有値分解（6問）
2. ベルヌーイ分布の平均、分散、負の対数尤度関数（3問）
3. ベイズの定理（3問）
4. KLダイバージェンス、交差エントロピー（3問）
5. 二乗和誤差、対数尤度関数（4問）
6. 機械学習（4問）
7. 汎化誤差、ホールドアウト法、交差検証法、ハイパーパラメーター、グリッドサーチ（5問）



## 8. k-means実装（5問）

- argがつくとindexを返す
- 関数名の意味を理解してればピンとくる？

cumulative

'kyōomyələdiv

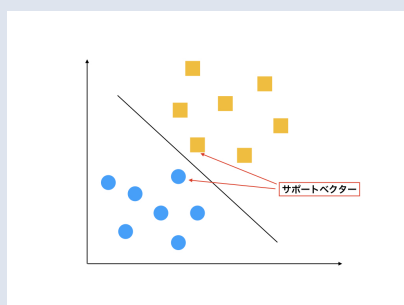
×

累積的な

Ruiseki-tekina

## 9. サポートベクトルマシン（3問）


- 外れ値の影響受けにくい ロバスト



## 10. シグモイド関数（4問）

## 11. 正則化

- 式覚える
- スパースの意味

 **sparse**

形

1. 〔物・人間・動物・植物などの存在量が少なくて〕まばらな、わずかな、希薄な◆【語源】  
「まき散らす」を意味するラテン語spargereの過去分詞sparsus  
• She has only sparse subcutaneous fat. : 彼女には、ほんのわずかな皮下脂肪しかない。  
• The backyard was flat with a sparse growth of grass. : 裏庭は草がまばらに生えていて平らだった。
2. 〔毛髪が〕薄い

## 12. 確率的勾配降下法の実装（4問）

- `train_size // batch_size` の// は、割り算の結果を小数点以下切り捨てて整数値を返す。
- numpyの出そうな関数まとめる
- パラメーター更新式を覚える。

## 13. Adam 実装（4問）

- 式覚える

$$m_{t+1} = \rho_1 m_t + (1 - \rho_1) \frac{\partial L}{\partial \theta_t}$$
$$v_{t+1} = \rho_2 v_t + (1 - \rho_2) \frac{\partial L}{\partial \theta_t} \odot \frac{\partial L}{\partial \theta_t}$$

$$\hat{m}_{t+1} = \frac{m_{t+1}}{1 - \rho_1^t}$$
$$\hat{v}_{t+1} = \frac{v_{t+1}}{1 - \rho_2^t}$$

$$\theta_{t+1} = \theta_t - \eta \frac{1}{\sqrt{\hat{v}_{t+1} + \varepsilon}} \odot \hat{m}_{t+1}$$

14. バッチ正規化（7問）

15. 勾配降下法（5問）

16. 畳み込みとプーリング（2問）

17. im2col実装（4問）

18. 代表的CNNモデル（3問）

19. 物体検出モデル（3問）

20. セマンティックセグメンテーション（1問）

21. 畳み込みのパラメータ（3問）

22. GRU (5問)

23. RNN 派生モデル (2問)

24. word2vec (2問)

25. BLEU (1問)

26. VAE (1問)

27. GAN (1問)

28. DQN (2問)

**29. 方策勾配法、Sarsa、Q学習（3問）**

**30. 性能指標（1問）**

**31. GNMT（1問）**

**32. BERT（1問）**

**33. WaveNet（1問）**

**34. 軽量化（蒸留、剪定、量子化）（1問）**

**35. ディファインアンドラン、ディファインバイラン (1問)**

**36. GPU (1問)**

**37. 分散深層学習 (1問)**





# 固有方程式

$$\det(\lambda I - A) = 0$$

## 各要素の意味など

- $\det$  は行列式(determinant) を意味する。  
 $|\lambda I - A| = 0$  と表現される事もある。
- $\lambda$  は固有値
- $I$  は単位行列
- $A$  は $n$ 次正方行列

# 特異値分解

$$A = U\Sigma V^T$$

## 各要素の意味など

- $A$  は
- $U$  は
- $\Sigma$  は
- $V^T$  は
-

# ベルヌーイ分布

$$f(x; p) = p^x (1 - p)^{1-x}$$

## 各要素の意味など

- $x$  は成功か失敗を表す変数 ( $k$ で表されることもある?)
- $p$  は単一試行での成功確率
- $f(x; p)$  はベルヌーイ分布の**確率質量関数**で、パラメータ $p$ が与えられた時の変数 $x$ の関数という意味
  - $x = 1$  の場合、 $p^x = p$  となり  $(1 - p)^{1-x} = 1$  となるため、 $f(x; p) = p$
  - $x = 0$  の場合、 $p^x = 1$  となり  $(1 - p)^{1-x} = 1 - p$  となるため、 $f(x; p) = 1 - p$
- $p$ の $x$ 乗の様な表現はただのモデル化であり物理的プロセスや現象を直接表現しているわけではない? (「こうするとうまく表現できる」以上の深い意味はない?)

# ベルヌーイ分布の期待値

$$\begin{aligned}\mathbb{E}[X] &= \sum_{x=0}^1 xp^x(1-p)^{1-x} \\ &= p\end{aligned}$$

## 各要素の意味など

- $\mathbb{E}[X]$ は

# ベルヌーイ分布の分散

$$Var[X] = \mathbb{E}[(X - \mathbb{E}[X])^2]$$

$$Var[X] = \mathbb{E}[X^2] - \mathbb{E}[X]^2$$

$$= \sum_{x=0}^1 x^2 p^x (1-p)^{1-x} - p^2$$

$$= p - p^2$$

$$= p(1-p)$$

各要素の意味など

-

# マルチヌーイ分布（カテゴリ分布）

$$f(x; p) = \prod_{j=1}^k p_j^{x_j} \quad (\text{ただし、} \sum_{j=1}^k p_j = 1, \quad 0 \leq p_j \leq 1, \quad j = 1, \dots)$$

各要素の意味など

-

# マルチヌーイ分布の負の対数尤度（カテゴリ分布）

$$\begin{aligned} -\log L_D(p) &= -\log \prod_{i=1}^n f(x_i; p) \\ &= -\sum_{i=1}^n \log \prod_{j=1}^k p_j^{x_{ij}} \\ &= -\sum_{i=1}^n \sum_{j=1}^k \log p_j^{x_{ij}} \\ &= -\sum_{i=1}^n \sum_{j=1}^k x_{ij} \log p_j \end{aligned}$$

各要素の意味など

-



# 正規分布

$$f(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp \left( -\frac{1}{2\sigma^2} (x - \mu)^2 \right)$$

各要素の意味など

-

# 正規分布の負の対数尤度

$$\begin{aligned} L(\mu) &= \prod_{i=1}^n f(x_i; \mu) \\ &= \prod_{i=1}^n \frac{1}{\sqrt{2\pi}} \exp \left( -\frac{1}{2} (x_i - \mu)^2 \right) \\ -\log L(\mu) &= -\log \left( \prod_{i=1}^n \frac{1}{\sqrt{2\pi}} \exp \left( -\frac{1}{2} (x_i - \mu)^2 \right) \right) \\ &= -\sum_{i=1}^n \log \left( \frac{1}{\sqrt{2\pi}} \exp \left( -\frac{1}{2} (x_i - \mu)^2 \right) \right) \\ &= -\sum_{i=1}^n \left( \log \left( \frac{1}{\sqrt{2\pi}} \right) - \frac{1}{2} (x_i - \mu)^2 \right) \\ &= -n \log \left( \frac{1}{\sqrt{2\pi}} \right) - \frac{1}{2} \sum_{i=1}^n (x_i - \mu)^2 \end{aligned}$$

各要素の意味など

•

a

# 正規分布の最尤推定

$$\begin{aligned}\frac{d}{d\mu}g(\mu) &= \frac{1}{2} \sum_{i=1}^n \frac{d}{d\mu} (x_i - \mu)^2 \\ &= \frac{1}{2} \sum_{i=1}^n (-2(x_i - \mu)) \\ &= \sum_{i=1}^n \mu - \sum_{i=1}^n x_i \\ &= n\mu - \sum_{i=1}^n x_i \\ \hat{\mu} &= \frac{1}{n} \sum_{i=1}^n x_i\end{aligned}$$

各要素の意味など

-

# エントロピー

$$H(X) = - \sum_x p(x) \log_2 p(x)$$

各要素の意味など

-

# 交差エントロピー(クロスエントロピー)の定義

$$H(p, q) = - \sum_x p(x) \log_2 q(x)$$

## 各要素の意味など

- $p(x)$  真の(正解の)確率分布
- $q(x)$  推定したモデルの確率分布

# 二値交差エントロピー(バイナリクロスエントロピー) ※1/8追加※

$$D_{BC} = -P(x = 0) \log Q(x = 0) - (1 - P(x = 0)) \log(1 - Q(x = 0))$$

## 各要素の意味など

- $p(x)$  真の(正解の)確率分布
- $q(x)$  推定したモデルの確率分布

# KLダイバージェンス

$$D(p||q) = \sum_x p(x) \log_2 \frac{p(x)}{q(x)}$$

各要素の意味など

-

# JSダイバージェンス

$$D_{JS}(p||q) = \frac{1}{2} \left( \sum_x p(x) \log_2 \frac{p(x)}{r(x)} + \sum_x q(x) \log_2 \frac{q(x)}{r(x)} \right)$$
$$r(x) = \frac{p(x) + q(x)}{2}$$

各要素の意味など

-



# ベイズの定理

$$p(C|x) = \frac{p(x|C)p(C)}{p(x)}$$

各要素の意味など

-

# バイアス・バリエーション・ノイズ

$$\begin{aligned}\mathbb{E}(L) &= \int \{y(x) - h(x)\}^2 p(x) dx + \iint \{h(x) - t\}^2 p(x, t) dx dt \\ &\quad \int \{\mathbb{E}_D[y(x; D)] - h(x)\}^2 p(x) dx \\ &\quad \int \mathbb{E}_D[\{y(x; D) - \mathbb{E}_D[y(x; D)]\}^2] p(x) dx \\ &\quad \iint \{h(x) - t\}^2 p(x, t) dx dt\end{aligned}$$

各要素の意味など

-

# オッズ

$$\frac{p(y = 1|x)}{p(y = 0|x)} = \frac{\hat{y}}{1 - \hat{y}}$$

$$\begin{aligned} \frac{\hat{y}}{1 - \hat{y}} &= \frac{\frac{1}{1 + \exp(-w^T x - b)}}{1 - \frac{1}{1 + \exp(-w^T x - b)}} \\ &= \frac{1}{(1 + \exp(-w^T x - b)) - 1} \\ &= \frac{1}{\exp(-w^T x - b)} \\ &= \exp(w^T x - b) \end{aligned}$$

各要素の意味など

-

# ガウスカーネル

$$k(x, x') = \exp \left( -\frac{\|x - x'\|^2}{\beta} \right)$$

各要素の意味など

-

# 正則化

$$E + \lambda_2 \|w\|_2^2$$

$$E + \lambda_1 \|w\|_1$$

$$E + \lambda_1 \|w\|_1 + \lambda_2 \|w\|_2^2$$

各要素の意味など

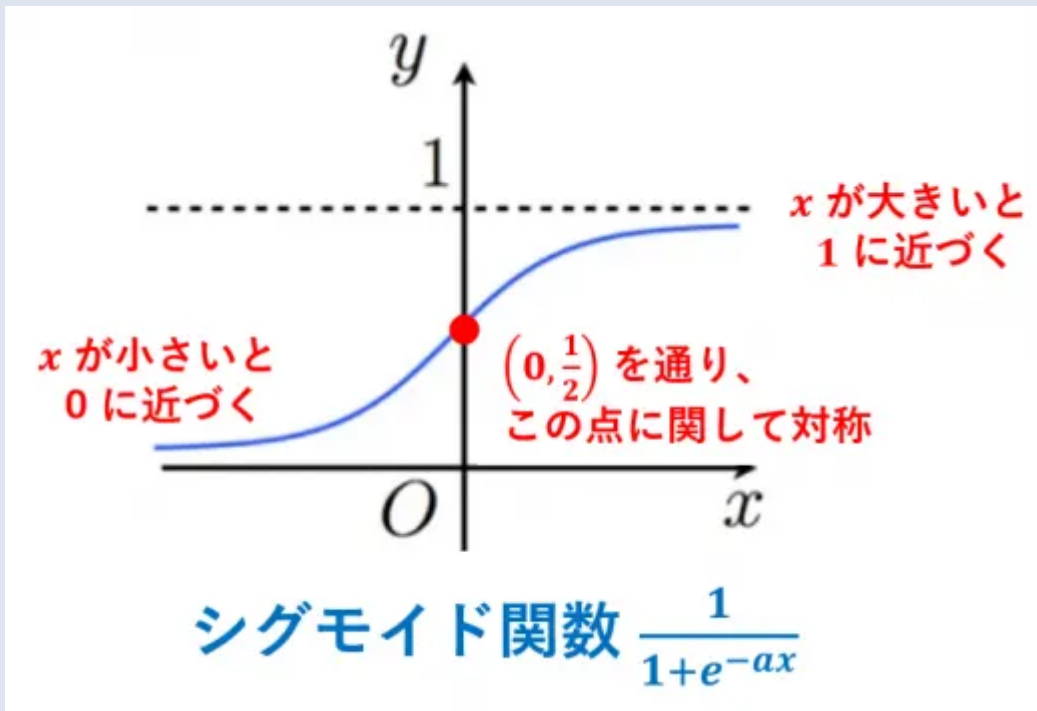
-

# シグモイド関数

$$f(x) = \frac{1}{1 + \exp(-x)}$$

## 各要素の意味など

- 初期の活性化関数
- 微分しやすい
- 層数が多いNNでは勾配消失が起こりやすいため、近年あまり使われていない
- どんな入力値に対しても  $0 < y < 1$  の範囲をとる
- $\exp(-x)$  は  $e^{-x}$  の意、 $x=0$ の時1となるため、 $f(x) = \frac{1}{2}$ となる



# ReLU関数

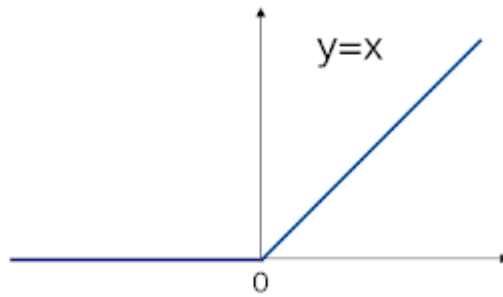
$$f(x) = \max(0, x)$$

```
np.maximum(0, x)
```

## 各要素の意味など

- ニューラルネットワークの活性化関数
- $\max(0, x)$ なので、 $y=x$ と $y=0$ のうち大きい方という意味
- $x=0$ の時は微分できない

ReLU  
Rectified  
Linear  
Unit



# ソフトマックス関数

$$\text{softmax}(z)_i = \frac{\exp(z_i)}{\sum_j \exp(z_j)}$$

$$y_i = \frac{e^{x_i}}{e^{x_1} + e^{x_2} + \dots + e^{x_n}} \quad (i = 1, \dots, n)$$

```
np.exp(z) / np.sum(np.exp(z))
```

## 各要素の意味など

- 分類問題の確率分布を表す為に使用
- $i=1 \sim n$ の場合それぞれの出力値の合計が1になる



## 二乗和誤差

$$\frac{1}{2} \sum_{k=1}^K (y_k - t_k)^2$$

$y_k$ で偏微分すると、 $k=1 \sim K$ のうち $k$ 番目以外の項が0になるため

$$y_k - t_k$$

### 各要素の意味など

- 損失関数の1つ
- $y_k - t_k$ は予測値と正解値との誤差。二乗することで必ず正の値になるようにしている。

# 生成モデル

$$p(y|x)p(x) = \frac{p(x, y)}{p(x)} \cdot p(x) = p(x, y)$$

## 各要素の意味など

-

# ベルマン方程式

$$V^\pi(s) = \mathbb{E}[G_t | S_t = s]$$

$$= \mathbb{E}[R_{t+1} + \gamma G_{t+1} | S_t = s]$$

$$= \sum_a \pi(a|s) \sum_{s', r} P(s', r|s, a) [r + \gamma \mathbb{E}_\pi[G_{t+1} | S_{t+1} = s']]$$

$$= \sum_a \pi(a|s) \sum_{s', r} P(s', r|s, a) [r + \gamma V^\pi(s')]$$

$$\begin{aligned} Q^\pi(s, a) &= \mathbb{E}[R_{t+1} + \gamma V^\pi(S_{t+1}) | S_t = s, A_t = a] \\ &= \sum_{s', r} P(s', r|s, a) [r + \gamma V^\pi(s')] \end{aligned}$$

各要素の意味など

-

# SARSA

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha[R_{t+1} + \gamma Q(S_{t+1}, A_{t+1}) - Q(S_t, A_t)]$$

各要素の意味など

-

# Q学習

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha \left[ R_{t+1} + \gamma \max_{a'} Q(S_{t+1}, a') - Q(S_t, A_t) \right]$$

各要素の意味など

-

# 方策勾配定理

$$\nabla_{\theta} J(\theta) = \sum d^{\pi_{\theta}}(s) \sum \nabla_{\theta} \pi_{\theta}(a|s, \theta) Q^{\pi_{\theta}}(s, a)$$

$$\nabla_{\theta} \log \pi_{\theta}(a|s) = \frac{\partial \pi_{\theta}(a|s)}{\partial \theta} \frac{1}{\pi_{\theta}(a|s)}$$

$$d^{\pi_{\theta}}(s) = \sum_{k=0}^{\infty} \gamma^k P^{\pi_{\theta}}(s_k = s | s_0)$$

$$\begin{aligned} \nabla_{\theta} J(\theta) &= \sum d^{\pi_{\theta}}(s) \sum_a (\nabla_{\theta} \pi_{\theta}(a|s, \theta)) Q^{\pi_{\theta}}(s, a) \\ &= \sum d^{\pi_{\theta}}(s) \sum_a \pi_{\theta}(a|s, \theta) (\nabla_{\theta} \log \pi_{\theta}(a|s, \theta)) Q^{\pi_{\theta}}(s, a) \\ &= \mathbb{E}_{\pi_{\theta}} [\nabla_{\theta} \log \pi_{\theta}(a|s, \theta) Q^{\pi_{\theta}}(s, a)] \end{aligned}$$

各要素の意味など

-

# モンテカルロ近似

$$\nabla_{\theta} J(\theta) = \mathbb{E}_{\pi_{\theta}}[f(s, a)] \approx \frac{1}{N} \sum_{n=1}^N \frac{1}{T} \sum_{t=1}^T \nabla_{\theta} \log \pi_{\theta}(a_t^n | s_t^n) Q^{\pi_{\theta}}(s_t^n, a_t^n)$$

各要素の意味など

-

## 交差エントロピー誤差（多クラス分類）

$$E = -\frac{1}{N} \sum_n \sum_k t_{nk} \log y_{nk}$$

### 各要素の意味など

- $N$  : データ個数（サンプル数）
- $k$  : データの次元数（クラス数？）



# アフィンレイヤ

$$H = XW + B$$

コード

```
# アフィン変換
affine[i] = np.dot(activations[i], self.coefs_[i]) + self.intercepts_[i]
```

重みの勾配

$$\frac{\partial L}{\partial W} = X^T \frac{\partial L}{\partial H}$$

入力の勾配

$$\frac{\partial L}{\partial X} = \frac{\partial L}{\partial H} W^T$$

```
# 入力(activations)の勾配を算出
deltas[i - 1] = np.dot(deltas[i], self.coefs_[i].T)
```

各要素の意味など

- X: 入力データ。
- W: 重み行列。
- B: バイアスベクトル。

# 確率的勾配降下法（SGD）

パラメータ更新式

$$\theta_{t+1} = \theta_t - \eta \frac{\partial L}{\partial \theta_t}$$

```
params[key] = params[key] - self.lr * grads[key]
```

# -=を使って下記の様に短く書ける

```
params[key] -= self.lr * grads[key]
```

各要素の意味など

- $\eta$ は学習率
- $\theta$ はパラメータ、 $\theta_t$ は更新前 $\theta_{t+1}$ が更新後
- $\frac{\partial L}{\partial \theta_t}$ はパラメータの勾配

# モーメンタム

- 更新速度の項を導入する事で勾配降下法よりも、谷での収束が早くなる
  - 同じ方向に進もうとする → 加速する
  - 逆の方向に進もうとする → ブレーキをかける

$$v_{t+1} = \alpha v_t - \eta \frac{\partial L}{\partial \theta_t}$$

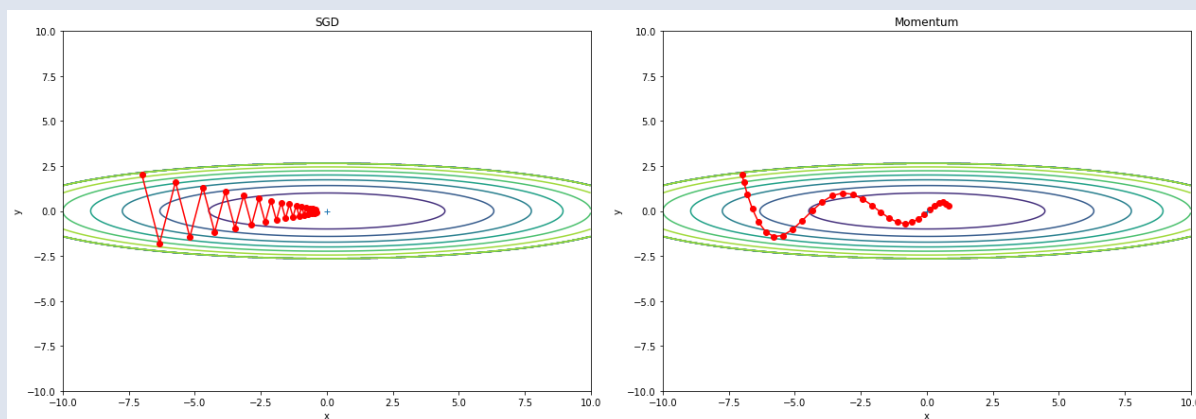
$$\theta_{t+1} = \theta_t + v_{t+1}$$

```
self.v[key] = self.momentum*self.v[key] - self.lr*grads[key]
params[key] += self.v[key]
```

## 各要素の意味など

- $\alpha$  : モーメンタム係数
- $v_t$  : 時刻tにおける速度 (パラメータの更新量)
- $\eta$  は学習率
- $\theta$  はパラメータ、 $\theta_t$  は更新前  $\theta_{t+1}$  が更新後
- $\frac{\partial L}{\partial \theta_t}$  はパラメータの勾配

## SGDとの比較



# NesterovAG (※モーメント改良版)

$$v_{t+1} = av_t - \eta \frac{\partial L}{\partial(\theta_t + av_t)}$$

$$\theta_{t+1} = \theta_t + v_{t+1}$$

計算難しいので実装し易くする

$$v_{t+1} = av_t - \eta \frac{\partial L}{\partial \Theta_t}$$

$$\Theta_{t+1} = \Theta_t + \alpha^2 v_t - (1 + \alpha) \eta \frac{\partial L}{\partial \Theta_t}$$

```
params[key] += self.momentum * self.momentum * self.v[key]
params[key] -= (1 + self.momentum) * self.lr * grads[key]
self.v[key] *= self.momentum
self.v[key] -= self.lr * grads[key]
```

## 各要素の意味など

- $\alpha$  : モーメント係数
- $v_t$  : 時刻tにおける速度 (パラメータの更新量)
- $\eta$  は学習率
- $\theta$  はパラメータ、 $\theta_t$  は更新前  $\theta_{t+1}$  が更新後
- $\frac{\partial L}{\partial \theta_t}$  はパラメータの勾配
-

# AdaGrad

- 勾配情報の蓄積により学習率を変化させる
- 過去の勾配情報を（アダマール積により）パラメータ毎に蓄積

$$h_{t+1} = h_t + \frac{\partial L}{\partial \theta_t} \odot \frac{\partial L}{\partial \theta_t}$$

$$\theta_{t+1} = \theta_t - \eta \frac{1}{\varepsilon + \sqrt{h_{t+1}}} \odot \frac{\partial L}{\partial \theta_t}$$

```
self.h[key] += grads[key] * grads[key]
params[key] -= self.lr * grads[key] / (np.sqrt(self.h[key]) + 1e-7)
```

## 各要素の意味など

- $h$  : 勾配情報を蓄積する変数
- $\varepsilon$  : 0での割り算を防ぐ小さな係数
- $\eta$  は学習率
- $\theta$  はパラメータ、 $\theta_t$  は更新前  $\theta_{t+1}$  が更新後
- $\frac{\partial L}{\partial \theta_t}$  はパラメータの勾配
- $\frac{\partial L}{\partial \theta_t} \odot \frac{\partial L}{\partial \theta_t}$  と二乗する事により、勾配が振動している時に  $h_{t+1}$ （第二式の分母）が大きくなるため、学習率を効果的に下げる事ができる。

# RMSPProp

- AdaGradの進化系、減衰率 $\rho$ の導入により、過去の勾配情報を「ある程度忘れる」ことができる。

$$h_{t+1} = \rho h_t + (1 - \rho) \frac{\partial L}{\partial \theta_t} \odot \frac{\partial L}{\partial \theta_t}$$

$$\theta_{t+1} = \theta_t - \eta \frac{1}{\sqrt{\varepsilon + h_{t+1}}} \odot \frac{\partial L}{\partial \theta_t}$$

```
self.h[key] *= self.decay_rate
self.h[key] += (1 - self.decay_rate) * grads[key] * grads[key]
params[key] -= self.lr * grads[key] / (np.sqrt(self.h[key] + 1e-7))
```

## 各要素の意味など

- $\rho$  : 減衰率 (decay rate)過去の勾配情報と現在の勾配情報の優先度合を決める係数
- $h$  : 勾配情報を蓄積する変数
- $\varepsilon$  : 0での割り算を防ぐ小さな係数
- $\eta$ は学習率
- $\theta$ はパラメータ、 $\theta_t$ は更新前 $\theta_{t+1}$ が更新後
- $\frac{\partial L}{\partial \theta_t}$ はパラメータの勾配

# Adam

$$\begin{aligned}
 m_{t+1} &= \rho_1 m_t + (1 - \rho_1) \frac{\partial L}{\partial \theta_t} \\
 v_{t+1} &= \rho_2 v_t + (1 - \rho_2) \frac{\partial L}{\partial \theta_t} \odot \frac{\partial L}{\partial \theta_t} \\
 \hat{m}_{t+1} &= \frac{m_{t+1}}{1 - \rho_1^t} \\
 \hat{v}_{t+1} &= \frac{v_{t+1}}{1 - \rho_2^t} \\
 \theta_{t+1} &= \theta_t - \eta \frac{1}{\sqrt{\hat{v}_{t+1}} + \varepsilon} \odot \hat{m}_{t+1}
 \end{aligned}$$

```

self.m[key] = self.rho1*self.m[key] + (1-self.rho1)*grads[key]
self.v[key] = self.rho2*self.v[key] + (1-self.rho2)*(grads[key]**2)

m = self.m[key] / (1 - self.rho1**self.iter)
v = self.v[key] / (1 - self.rho2**self.iter)

params[key] -= self.lr * m / (np.sqrt(v) + self.epsilon)

```

## 各要素の意味など

- $m$  : 更新速度を表す変数、Adamではここにも減衰率を導入している。
  - モーメンタムでは速度が $v$ だったのにややこしい！
- $v$  : 勾配情報を蓄積する変数、RMSpropの $h$ と同じ
- $\hat{m}$ および $\hat{v}$  : 上記 $m$ 、 $v$ は偏差を含むため、それを補正しているらしい。
  - 初期値が0から始まるため学習初期段階でのモーメント推定が実際の値よりも小さく偏ってしまう。それを補正するのが目的。
  - 学習が進む（ $t$  : イテレーション数が増える）と分母が1に近づくため、補正の効果が小さくなる。

# バッチ正規化

- ミニバッチ毎や、学習データとテストデータで特徴量の分布に違いがあることがあるため学習が安定しない
- ミニバッチ毎に各チャンネルを平均0分散1にする

$$h' = \frac{h - \mu}{\sigma}$$

- モデルの表現力維持のためにスケーリングとシフトを行う

$$\gamma h' + \beta$$

## 各要素の意味など

- $h$  : バッチ正規化層のあるノードへの入力
- $\mu$  : 平均
- $\sigma$  : 標準偏差
- $\gamma$  : スケーリング係数（学習されるパラメータ）
- $\beta$  : シフト係数（学習されるパラメータ）

## 利点

- 学習率を高く設定できる（パラメータのスケールに依存しなくなる）
- 正則化の効果がある
- 訓練が速く進む（損失値がはやく収束する）
- 重みの初期化に神経質にならなくとも良い



# 畳み込み

$$(I * K)(i, j) = \sum_m \sum_n I(i + m, j + n) K(m, n)$$

各要素の意味など

-

# IoU

$$IoU(B_{true}, B_{pred}) = \frac{|B_{true} \cap B_{pred}|}{|B_{true} \cup B_{pred}|} = \frac{|B_{true} \cap B_{pred}|}{|B_{true}| + |B_{pred}| - |B_{true} \cap B_{pred}|}$$

## 各要素の意味など

-

# Dice 係数

$$Dice(S_{true}, S_{pred}) = \frac{|S_{true} \cap S_{pred}|}{\frac{|S_{true}| + |S_{pred}|}{2}} = \frac{2|S_{true} \cap S_{pred}|}{|S_{true}| + |S_{pred}|}$$

各要素の意味など

-

$$AP = \frac{1}{11} \sum_{r \in \{0, 0.1, 0.2, \dots, 1\}} p_{interp}(r)$$

$$p_{interp}(r) = \max_{\tilde{r} \geq r} p(\tilde{r})$$

各要素の意味など

-

# LSTMの順伝播

$$G = \tanh \left( X_t W_x^{(g)} + H_{t-1} W_h^{(g)} + b^{(g)} \right)$$

$$I = \text{sigmoid} \left( X_t W_x^{(i)} + H_{t-1} W_h^{(i)} + b^{(i)} \right)$$

$$F = \text{sigmoid} \left( X_t W_x^{(f)} + H_{t-1} W_h^{(f)} + b^{(f)} \right)$$

$$O = \text{sigmoid} \left( X_t W_x^{(o)} + H_{t-1} W_h^{(o)} + b^{(o)} \right)$$

各要素の意味など

-

## GRUの順伝播

$$R = \text{sigmoid} \left( X_t W_x^{(r)} + H_{t-1} W_h^{(r)} + b^{(r)} \right)$$

$$Z = \text{sigmoid} \left( X_t W_x^{(z)} + H_{t-1} W_h^{(z)} + b^{(z)} \right)$$

$$\tilde{H} = \tanh \left\{ X_t W_x^{(\tilde{h})} + (R \odot H_{t-1}) W_h^{(\tilde{h})} + b^{(\tilde{h})} \right\}$$

$$H_t = Z \odot H_{t-1} + (1 - Z) \odot \tilde{H}$$

各要素の意味など

-

# WaveNetの定式化

$$p(x) = \prod_{t=1}^T p(x_t | x_1, x_2, \dots, x_{t-1})$$

各要素の意味など

-

# TransformerのScaled Dot-Product Attention

$$\text{Attention}(Q, K, V) = \text{softmax} \left( \frac{QK^T}{\sqrt{d_k}} \right) V$$

各要素の意味など

-



# TransformerのPositional Encoding

$$PE_{(pos, 2i)} = \sin \left( pos / 10000^{2i/d_{model}} \right)$$

$$PE_{(pos, 2i + 1)} = \cos \left( pos / 10000^{2i/d_{model}} \right)$$

各要素の意味など

-

# VAEの損失関数

$$-\log p(x) \leq -L = \mathbb{E}_{z \sim p(z|x)} [-\log p(x|z)] + \int \log \left( \frac{p(x|z)}{p(z)} \right) p(z|x) dz$$

各要素の意味など

-

# GANの定式化

$$\min_G \max_D \mathbb{E}_{\mathbf{x}} [\log D(x)] + \mathbb{E}_{\mathbf{z}} [\log(1 - D(G(z)))]$$

各要素の意味など

-

# DQN

$$L(\theta) = \mathbb{E}_{s,a,r,s' \sim \mathbb{D}} \left[ \left( r + \gamma \max_{a'} Q(s', a'; \theta^-) - Q(s, a; \theta) \right)^2 \right]$$

## 各要素の意味など

-

# 蒸留における温度付きソフトマックス関数

$$\text{Softmax}(z)_i = \frac{\exp(z_i/T)}{\sum_j \exp(z_j/T)}$$

各要素の意味など

-