Ponieważ warstwa końcowa sieci może poznać przekształcenie liniowe, możemy chcieć usunąć wszystkie związki liniowe między jednostkami w obrębie warstwy. Jest to właśnie podejście przyjęte przez Desjardins et al. (2015), którzy dali inspirację do normalizacji pakietowej. Niestety eliminacja wszystkich liniowych interakcji jest znacznie bardziej kosztowna niż standaryzacja średniej arytmetycznej i odchylenia standardowego każdej pojedynczej jednostki, więc jak dotąd normalizacja pakietowa pozostaje najbardziej praktycznym podejściem.

Normalizacja średniej arytmetycznej i odchylenia standardowego jednostki może zmniejszyć moc ekspresji wyrażanej przez sieć neuronową zawierającą tę jednostkę. Aby utrzymać tę moc, często zastępuje się pakiet aktywacji ukrytych jednostek $\boldsymbol{H}$ przez $\boldsymbol{\gamma H'} + \boldsymbol{\beta}$ zamiast po prostu znormalizowanej $\boldsymbol{H'}$. Zmienne $\boldsymbol{\gamma}$ i $\boldsymbol{\beta}$ to poznane parametry, które pozwalają, aby nowa zmienna miała dowolną średnią arytmetyczną i odchylenie standardowe. Na pierwszy rzut oka może się to wydawać bezużyteczne – po co ustalaliśmy średnią arytmetyczną na $\mathbf{0}$, a następnie wprowadzaliśmy parametr, który pozwala cofnąć ją na dowolną arbitralną wartość $\boldsymbol{\beta}$? Ponieważ nowa parametryzacja może reprezentować tę samą rodzinę funkcji wejścia, jak stara parametryzacja, ale ta nowa ma inną dynamikę uczenia się. W starej parametryzacji średnia arytmetyczna $\boldsymbol{H}$ była zdeterminowana przez skomplikowane interakcje między parametrami w warstwach poniżej $\boldsymbol{H}$. W nowej parametryzacji średnia arytmetyczna $\boldsymbol{\gamma H'} + \boldsymbol{\beta}$ jest zdeterminowana wyłącznie przez $\boldsymbol{\beta}$. Nowa parametryzacja jest znacznie łatwiejsza do uczenia ze spadkiem gradientu.

Większość warstw sieci neuronowej przyjmuje postać $\phi(\boldsymbol{XW}(+\boldsymbol{b})$, gdzie ϕ jest pewną ustaloną nieliniową funkcją aktywacji, jak poprawione przekształcenie liniowe. Możemy oczywiście zastanawiać się, czy należy stosować normalizację pakietową dla wejścia $\boldsymbol{X}$ lub do przekształconej wartości $\boldsymbol{XW}+\boldsymbol{b}$. Ioffe i Szegedy (2015) zalecają to drugie. Konkretniej $\boldsymbol{XW} + \boldsymbol{b}$ powinno zostać zastąpione przez znormalizowaną wersję $\boldsymbol{XW}$. Składnik obciążenia powinien zostać pominięty, gdyż staje się on nadmiarowy przy zastosowaniu przez reparametryzację normalizacji pakietowej parametru β. Dane wejściowe do warstwy są zwykle wynikami nieliniowej funkcji aktywacji, jak poprawiona funkcja liniowa z poprzedniej warstwy. Statystyki dla wejścia są więc mniej gaussowskie i mniej podatne na standaryzację przez działania liniowe.

W sieciach splotowych opisanych w rozdziale 9 ważne jest zastosowanie tej samej normalizacji μ i σ w każdej lokalizacji przestrzennej na odwzorowaniu cech, tak aby statystyki odwzorowania pozostały takie same niezależnie od położenia przestrzennego.

8.7.2. Spadek współrzędnych

W niektórych przypadkach można szybko rozwiązać problem optymalizacyjny, dzieląc go na oddzielne fragmenty. Jeśli minimalizujemy $f(\boldsymbol{x})$ względem jednej zmiennej x_i, następnie minimalizujemy ją względem innej zmiennej x_j itd., powtarzając to cyklicznie dla wszystkich zmiennych, to mamy pewność dojścia do minimum (lokalnego). Ta praktyka jest znana jako **spadek współrzędnych**, gdyż optymalizujemy współrzędne po jednej. Uogólniając, **blokowy spadek współrzędnych** odnosi się do jednoczesnej minimalizacji względem podzbioru zmiennych. Określenie „spadek współrzędnych" jest często używane w odniesieniu do blokowego spadku współrzędnych, a także do ściśle indywidualnego spadku współrzędnych.

Spadek współrzędnych jest najbardziej celowy, gdy różne zmienne w problemie optymalizacyjnym mogą być jasno podzielone na grupy, które grają względnie izolowane od siebie role, lub gdy optymalizacja względem jednej grupy zmiennych jest znacznie bardziej wydajna niż optymalizacja względem wszystkich zmiennych. Rozważmy dla przykładu funkcję kosztów:

$$J(\boldsymbol{H}, \boldsymbol{W}) = \sum_{i,j} |H_{i,j}| + \sum_{i,j} \left(\boldsymbol{X} - \boldsymbol{W}^\top \boldsymbol{H} \right)^2_{i,j}. \tag{8.38}$$

Ta funkcja opisuje problem uczenia się, określany jako rzadkie kodowanie, gdzie celem jest znalezienie macierzy wag $\boldsymbol{W}$, które mogą liniowo zdekodować macierz aktywacji wartości $\boldsymbol{H}$, aby dokonać rekonstrukcji zbioru szkoleniowego $\boldsymbol{X}$. Większość zastosować rzadkiego kodowania obejmuje też zanikanie wagi lub ograniczenie na normy kolumn $\boldsymbol{W}$, aby zapobiec patologicznym rozwiązaniom z niezwykle małymi wartościami $\boldsymbol{H}$ i dużymi $\boldsymbol{W}$.

Funkcja J nie jest wypukła. Jednak możemy podzielić wejścia do algorytmu szkoleniowego na dwa zbiory: parametry słownikowe $\boldsymbol{W}$ oraz reprezentacje kodu $\boldsymbol{H}$. Minimalizacja funkcji celu względem jednego z tych zbiorów zmiennych to problem wypukły. Blokowy spadek współrzędnych daje więc strategię optymalizacji, która pozwala nam na użycie skutecznego algorytmu optymalizacyjnego przez przełączenie się między optymalizacją W przy ustalonym $\boldsymbol{H}$, a potem optymalizację $\boldsymbol{H}$ przy ustalonym $\boldsymbol{W}$.

Spadek współrzędnych nie jest zbyt dobrą strategią, gdy wartość jednej ze zmiennych silnie wpływa na wartość optymalną innej zmiennej, jak ma to miejsce w funkcji $f(\boldsymbol{x}) = (x_1 - x_2)^2 + \alpha(x_1^2 + x_2^2)$, gdzie α jest dodatnią stałą. Pierwszy składnik zachęca obie zmienne do przyjęcia podobnych wartości, a drugi do przyjęcia wartości bliskich 0. Rozwiązaniem jest ustawienie obu na 0. Metoda Newtona może rozwiązać problem w jednym kroku, gdyż jest to dodatnio określony problem kwadratowy. Jednak dla małych wartości