An analysis on negative curvature induced by singularity in multi-layer neural-network learning

Authored by:

Eiji Mizutani
Stuart Dreyfus

Abstract

In the neural-network parameter space, an attractive field is likely to be induced by singularities. In such a singularity region, first-order gradient learning typically causes a long plateau with very little change in the objective function value E (hence, a flat region). Therefore, it may be confused with "attractive" local minima. Our analysis shows that the Hessian matrix of E tends to be indefinite in the vicinity of (perturbed) singular points, suggesting a promising strategy that exploits negative curvature so as to escape from the singularity plateaus. For numerical evidence, we limit the scope to small examples (some of which are found in journal papers) that allow us to confirm singularities and the eigenvalues of the Hessian matrix, and for which computation using a descent direction of negative curvature encounters no plateau. Even for those small problems, no efficient methods have been previously developed that avoided plateaus.

1 Paper Body

Consider a general two-hidden-layer multilayer perceptron (MLP) having a single (terminal) output, H nodes at the second hidden layer (next to the terminal layer), I nodes at the first hidden layer, and J nodes at the input layer; hence, a J-I-H-1 MLP. It has totally n parameters, denoted by an

n-vector ?, including thresholds. Let ?(.) be some node function; then, the forward pass transforms the input vector x of length J to the first hidden-output vector z of length I , and then to the second hidden-output vector h of length H , leading to?the final ? P output y: ? ?P ‘ T ? H H T with zk = ?(xT+ wk). y = f (?; x) = ? h+ p = ? (1) j=0 pj ?(z+vj) j=0 pj hj = ?

Here, fictitious outputs x0 = z0 = h0 = 1 are included in the output vectors with subscript ?+? for thresholds p0 , v0,j , and w0,k ; pj (j = 1, ..., H) is the weight connecting the jth hidden node to the (final) output; vj a vector of ?hidden? weights directly connecting to the jth hidden node from the first hidden layer; wk a?vector of ?hidden? weights to the kth hidden node from the input layer; ? ? ? T hence, ? T ? pT —vT —wT = pT —v1T , ..., vjT , ..., vH —w1T , ..., wkT , ..., wIT . The length of those weight vectors ?, p, v, w are denoted by n, n3 , n2 , and n1 , respectively, where n = n3 +n2 +n1 ; n3 = (H +1); n2 = H(I +1); n1 = I(J +1). (2) For parameter optimization, one may attempt to minimize the squared error over m data m m 1X 1X 2 1 2 (3) E(?) = {f (?; xd)?td } = rd (?) = rT r, 2 2 2 d=1 d=1 where td is a desired output on datum d; each residual rd a smooth function from ¡n to ¡; and r an m-vector of residuals. Note here and hereafter that the argument (?) for E and r is frequently suppressed as long as no confusion arises. The gradient and Hessian of E can be expressed as below ?E(?) =

m X d=1

rd ?rd = JT r, and ?2 E(?) =

m X d=1

?rd ?rdT +

m X d=1

rd ?2 rd ? JT J+S,

where J ? ?r, an m?n Jacobian matrix of r, and the dth row of J is denoted by ?r dT . 1

(4)

In the well-known Gauss-Newton method, S, the last matrix of second derivatives of residuals, is omitted, and its search direction ?? is found by solving J?? GN = ?r (or, JT J??GN = ??E). Under the normal error assumption, the Fisher information matrix is tantamount to J T J, called the GaussNewton Hessian. This is why natural gradient learning can be viewed as an incremental version of the Gauss-Newton method (see p.1404 [1]; p.1031 [2]) in the nonlinear least squares sense. Since JT J is positive (semi)definite, natural gradient learning has no chance to exploit negative curvature. It would be of great value to understand the weaknesses of such Gauss-Newton-type methods. Learning behaviors of layered networks may be attributable to singularities [3, 2, 4]. Singularities have been well discussed in the nonlinear least squares literature also: For instance, Jennrich & Sampson (pp.65?66 [5]) described an overlap-singularity situation involving a redundant model; specifically, a classical (linear-output) model of exponentials with hi ? ?(vi x) and no thresholds in Eq.(1): f (?; x) = p1 ?(v1 x)+p2 ?(v2 x) = p1 ev1 x +p2 ev2 x . (5) If the target data follow the path of a single exponential then the two hidden parameters, v 1 and v2 , become identical (i.e., overlap singularity) at the solution point, where J is rank-deficient; hence, JT J is singular. If the fitted response function nearly follows such a path, then J T J is nearly singular. This is a typical over-realizable scenario, in which the true teacher lies at the singularity (see [6] for details about 1-2-1 MLP-learning). In practice, if the solution point ? ? is stationary but J(? ?) is rank-deficient, then the search direction ?? GN can be numerically orthogonal to ?E at some distant point from ? ? ; consequently, no progress can be made by searching along the Gauss-Newton direction (hence, line-search-based algorithms fail); this is first pointed out by Powell, who proved in [7] that the Gauss-Newton iterates converge to a non-stationary limit point at which J is rank-deficient in solving a particular system of nonlinear equations, for which the merit function is defined as Eq.(3), where m = n. Another weak point of the Gauss-Newton-type method is a so-called largeresidual problem (e.g., see Dennis [8]); this implies that S in ?2 E is substantial because r is highly nonlinear, or its norm is large at solution ? ? . Those drawbacks of the Gauss-Newton-type methods indicate that negative curvature often arises in MLP-learning when JT J is singular (i.e., in a rank-deficient nonlinear least squares problem), and/or when S is more dominant than J T J. We thus verify this fact mathematically, and then discuss how exploiting negative curvature is a good way to escape from singularity plateaus, thereby enhancing the learning capacity.

2

Negative curvature induced by singularity

In rank-deficient nonlinear least squares problems, where J ? ?r is rank deficient, negative curvature often arises. This is true with an arbitrary MLP model, but to make our P analysis concrete, we consider a single terminal linear-output two-hidden-layer MLP: f (?; x) = H j=0 pj hj in Eq. (1). Then, the n weights separate into linear p and non-linear v and w. In this context, we show that a 4-by-4 indefinite Hessian block can be extracted from the n-by-n Hessian matrix ? 2 E in Eq.(4). 2.1

An existence of the 4 ? 4 indefinite Hessian block H in ?2 E

In the posed two-hidden-layer MLP-learning, as indicated after Eq.(1), the n weights are organized ? ? as ? T ? pT —vT —wT . Now, we pay attention to two particular hidden nodes j and k at the second hidden layer. The weights connecting to those two nodes are pj , pk , vj , and vk ; they are arranged in the following manner: ? ? ? T = p0 , p1 , ..., pj , ..., pk , ..., pH —v0,1 , ..., —v0,j , v1,j , ..., vI,j —...—v0,k , v1,k , ..., vI,k —...., — wT , (6) where vi,k is a weight from node i at the first hidden layer to node k at the second hidden layer. Given a data pair (x; t), r ? f (?; x)?t, a residual element, and uT , an n-length row vector of the residual Jacobian matrix J (? ???r) in Eq.(4), is given as below using the output vector z+ (including z0 = 1) at the first hidden layer ? ? uT ? ?r T = ..., hj , ..., hk , ..., ?0j (zT+ vj)pj , ..., ?0k (zT+ vk)pk , ... , (7) where only four entries are shown that are associated with four weights: pj , pk , v0,j , and v0,k . The locations of those four weights in the n-vector ? are denoted by l1 , l2 , l3 , and l4 , respectively, where l1 ? j +1, l2 ? k+1, l3 ? (I +1)(j ?1)+1, l4 ? (I +1)(k?1)+1. (8) Given J, we interchange columns 1 and l1 ; then, do columns 2 and l2 ; then columns 3 and l3 ; and finally columns 4 and l4 ; this interchanging procedure moves those four columns to the first four. 2

Suppose that the n ? n Hessian matrix ?2 E = uuT +S is evaluated on a given single datum (x; t). We then apply the above interchanging procedure to both rows and columns of ? 2 E appropriately, which can be readily accomplished by PT ?2 E P, where four permutation matrices Pi (i = 1, ..., 4) are employed as P ? P1 P2 P3 P4 ; each Pi satisfies PTi Pi = I (orthogonal) and Pi = PTi (symmetric); hence, P is orthogonal. As a result, H, the 4-by-4 Hessian block (at the upper-left corner) of the first four leading rows and columns of PT ?2 E P has the following structure: 2

(hj)2

6 H =6 4 —{z} 4?4

hj hk (hk)2

Symmetric

3 2 hj ?0j (.)pj hj ?0k (.)pk 0 ?0j (.)r 0 0 0 hk ?j (.)pj hk ?k (.)pk 7 0 0 7+6 ? 0 ?2 0 4 ?00j (.)pj r ?j (.)pj ?j (.)?0k (.)pj pk 5 2 Symmetric {?0k (.)pk }

0 ?0k (.)r

3

7 5. 0 00 ?k (.)pk r

(9)

The posed Hessian block H is associated with a vector of the four weights [pj , pk , v0,j , v0,k]T . If vj = vk , then hj = hk = ?(zT+ v); see Eq.(7). Obviously, no matter how many data are accumulated, two columns hj and hk of J in Eq.(4) are identical; therefore, J is rank deficient; hence, JT J is singular. The posed singularity gives rise to negative curvature because the above 4-by-4 dense Hessian block is almost always indefinite (so is ?2 E of size n ? n) to be proved next. 2.2

Case 1: vj = vk ? v; hence, hj = hk ? h = ?(zT+ v), and pj 6= pk

Given a set of m (training) data, the gradient vector ?E and the Hessian matrix ?2 E in Eq.(4) are evaluated. We then apply the aforementioned orthogonal matrix P to them as PT ?E and PT ?2 EP, yielding the gradient vector g of length 4 and the 4-by-4 Hessian block H [see Eq.(9)] associated with the four weights [pj , pk , v0,j , v0,k]T ; they may be expressed in a compact form as ? ? ? ? ? ? b2 0 0 0 e a a b1 ? m X b ? ?0 0 0 e ? ? ? ? ? a a b g= rd ud = ? p e ?; H = JT J+S = ? b b c 1 c 2 ? + ? e 0 d , (10) 0 ? j

d=1

1

pk e

1

11

12

1

c22 0 e 0 d2 P Pm ? 0 T ?2 Pm 0 T 00 T where the entries are given below with B ? d=1 ? (z+dv)hd , C ? d=1 ? (z+dv) , D ? m d=1 ? (z+dv)rd : ? m X 2 ? ? hd , b1 ? pj B, b2 ? pk B, c11 ? p2j C, c12 ? pj pk C, c22 ? p2k C, ? ? a? b2

b2

c12

d=1

m m X X ? ? ? ? ? ? r h , e ? ?0 (zT+d v)rd , d1 ? pj D, d d ? d=1

d=1

(11)

d2 ? pk D.

Notice here that the subscript d implies datum d (d = 1, ..., m); hence, hd is the hidden-node output on datum d (but not the dth hidden-node output) common to both nodes j and k due to v j = vk = v. Theorem 1: When e 6= 0, the n-by-n Hessian ?2E and its block H in Eq.(10) are always indefinite. Proof:

A similarity transformation with T, a 4-by-4 orthogonal matrix (TT = T?1), obtains 2

2a b1 +b2 +e ? 6b1 +b2 +e T T HT = 4 0 0 b1 ?b2 ?

3 2 ?1 0 b1 ?b2 2 6 ?1 0 ? 7 6 2 with T = 6 4 0 ?12 e 5 0 ?12 ?

0 0 0 e

1 ? 2 ?1 ? 2

3 0 07 7 7, 0 ?12 5 ?1 ? 0 2

(12)

where ? ? 21 (c11+2c12+c22+d1+d2), ? ? 21 (c11?c22+d1?d2), and ? ? 12 (c11?2c12+c22+d1+d2). The eigenvalues of the 2-by-2 block at the lower-right corner are obtainable by ?

? h i? ? ? ??I ? 0e ?e ? = ?(? ? ?) ? e2 = ?2 ? ? ? ? e2 = 0,

which yields 21 (? ? ? 2 + 4e2), the ?sign-different? eigenvalues as long as e 6= 0 holds. Then, by Cauchy?s interlace theorem (see Ch.10 of Parlett 1998), the Hessian ?2E is indefinite. (So is H.) 2 2.3

Case 2: vj = vk ? v (hj = hk ? h), and pj = pk ? p

The result in Case 1 becomes simpler: For a given set of m (training) data, ? ? 3 2 2 ? 0 0 a a b b Pm ?? ? 6a a b b7 60 0 T + g = d=1 rd ud = ? ?; H = J J+S = 4 b b c c5 4 e 0 pe b b c c 0 e pe 3

e 0 d 0

3 0 e 7 , 05 d

(13)

where the entries are readily identifiable from Eq.(11). In Eq.(13), JT J is positive semi-definite (singular of rank 2 even when m ? 2), and S has an indefinite structure. When e 6= 0 (hence, ?E 6= 0), we can prove below that there always exists negative curvature (i.e., ?2 E is always indefinite). Theorem 2: When e 6= ?0, the 4?4 Hessian block H in Eq.(13) includes the sign-different eigenvalues of S; namely, 12 (d ? d2 + 4e2), and the n ? n Hessian ?2 E as well as H are always indefinite. Proof: Proceed similarly with the same orthogonal matrix T as defined in Eq.(12), where b 1 = b2 = b, ? = 0, and ? = d, rendering TT HT ?block-diagonal.? Its block of size 2 ? 2 at the lower-right corner has the sign-different eigenvalues determined by ?2 ?d??e2 = 0. 2 QED 2 Now, we investigate stationary points, where the n-length gradient vector ?E = 0; hence, g = 0 in Eq.(13). We thus consider two cases for pe = 0: (a) p = 0 and e 6= 0, and (b) p 6= 0 and e = 0. In Case (b), S becomes a diagonal matrix, and the above TT H T shows that H is of (at most) rank 3 (when d 6= 0); hence, H becomes singular. Theorem 3: If ?E(? ?) = 0, p = 0, and e 6= 0 [i.e., Case (a)], then the stationary point ? ? is a saddle. Theorem 4: If ?E(? ?) = 0, and e = 0, but d ¡ 0 [see Eq.(13)], then ? ? is a saddle point. Proof of Theorems 3 and 4: From Theorem 2 above, H in Eq.(13) has a negative eigenvalue; hence, the entire Hessian matrix ?2 E of size n ? n is indefinite 2 QED 2

Theorem 4 is a special case of Case (b). If d = pD ¿ 0, then H becomes positive semi-definite; however, we could alter the eigen-spectrum of H by changing linear parameters p in conjunction with scalar ? for pj = 2?p and pk = 2(1??)p such that pj +pk = 2p with no change in E and ?E = 0 held fixed (to be confirmed in simulation; see Fig.1 later), leading to the following Theorem 5: If D

6= 0 and C ¿ 0 [see the definition of C and D for Eq.(11)] and v1 = v2 (? v) with ?E = 0, for which p 6= 0 and e = 0 (hence, S is diagonal), then choosing scalar ? appropriately for pj = 2?p and pk = 2(1??)p can render H and thus ?2 E indefinite. Proof: From Eq.(11), two on-diagonal (3,3) and (4,4) entries of H are a quadratic function in terms D of ?: The (3,3)-entry of H, H(3, 3) = 2?p(2?pC + D), has two roots: 0 and ? 2pC , whereas the D (4,4)-entry, H(4, 4) = 2(1??)p[2(1??)pC+D], has two roots: 1 and 1 + 2pC . Obviously, given p, C, and D, there exists ? such that the quadratic function value becomes negative (see later Fig.1). This implies that adjusting ? can produce a negative diagonal entry of H; hence, indefinite. Then, again by Cauchy?s interlace theorem, so is ?2 E . 2 QED 2 Example 1: A two-exponential model in Eq.(5). Data set 1:

Input x	Target t
?2	1
?1	3
0	2
1	3
2	1

Data set 2:

Input x	Target t
?2	3
?1	1
0	2
1	1
2	3

(14)

Given two sets of five data pairs (xi ; ti) as shown above, for each data set, we first find a minimizer ? 0? = [p? , v?]T of a two-weight 1-1-1 MLP, and then expand it with scalar ? as ? = [?p? , (1 ? ?)p? , v? , v?]T to construct a four-weight 1-2-1 MLP that produces the same input-tooutput relations. That is, we first find the minimizer ? 0? = [p? , v?]T using a 1-1-1 MLP, f (? 0 ; x) = pevx , ?2 ? P by solving ?E = 0, which yields p? = 2; v? = 0; E(? 0?) = 21 5j=1 f (? 0? ; xj) ? tj = 2; and confirm that the 2 ? 2 Hessian ?2 E(? 0?) is positive definite in both data sets above. Next, we augment ? 0? as ? = [p1 , p2 , v1 , v2]T = [?p? , (1 ? ?)p? , v? , v?]T to construct a 1-2-1 MLP: f (?; x) = p1 ev1 x +p2 ev2 x , which realizes the same input-to-output relations as the 1-1-1 MLP. Fig.1 shows how ? changes the eigen-spectrum (see solid curve) of the 4 ? 4 Hessian ?2 E (supported by Theorem 5).

Conjecture: Suppose that ? ? is a local minimum point in two-hidden-layer J-I-H-1 MLP-learning, and ?2 E of size n ? n is positive definite (so is H) with ?E = 0 and E ¿ 0. Then, adding a node at the second hidden layer can increase learning capacity in the sense that E can be further reduced. Sketch of Proof: Choose a node j among H hidden nodes, and add a hidden node (call node k) by duplicating the hidden weights by vk = vj with pk = 0; hence, totally n e ? n+(I +2) weights. This certainly renders new JT J of size n e?n e singular, and the (4,4)-entry in H in Eq.(10) becomes zero (due to pk = 0). Then, by the interlace theorem, new ?2 E of size n e?n e becomes indefinite. 2

The above proof is not complete since we did not make clear assumptions about how the first-order necessary condition ?E = 0 holds [see Cases (a) and (b) just above Theorem 3]. Furthermore, even if we know in advance the minimum number of hidden nodes, Hmin , for a certain task, we may not be able to find a local-minimum point of an MLP with one less hidden nodes, H min ?1. Consider, for instance, the well-known (four data) XOR problem. Although it can be solved by a 2-2-1 MLP (nine 4

20
20
15
15
10
10
5
5
0
0
?5
?5
?10
min Eig(? E) min Eig(S) ?2E(3,3) 2 ? E(4,4)
?15 ?20 ?25 ?2
?1
0
?
2
min Eig(? E) min Eig(S) 2 ? E(3,3) 2 ? E(4,4)
?10
2
?15 ?20
1
2
?25 ?2
3
?1
0
1
?
2
3
2

Figure 1: The change of the minimum eigenvalue of ? E (solid curve) and of S (dashed) as well as the (3,3)-entry of ?2 E (dotted) and the (4,4)-entry of ?2 E (dash-dot), both quadratic, according to value ? (x-axis) in ? = [?p? , (1??)p? , v? , v?]T , the four weights of a 1-2-1 MLP with exponential hidden nodes

(left) using data set 1, and (right) data set 2 in Eq.(14). Theorem 5 supports this result. 2?D contour plot 2?D contour plot

5 20
4 15
Minimizer
3
Attractor
10
Minimizer 4
2
Minimizer
5
Saddle
0
E(p,v)
v
v
3 1
Saddle 0
?5
?1
?10
2 1
0 2.5
Attractive point
?20 2
?3
0
0.2
0.4
1.5
?15
Saddle
?2
?10 1
0.6
0.8
1
1.2
?20 ?1
?0.5
0
p
x
0.5
1

1.5
2
2.5
p
p
0 0.5
v
10
0 ?0.5 ?1
20

(a) (b) (c) Figure 2: The 1-1-1 MLP landscape: (a) a magnified view; (b) bird?s-eye views in 2-D, and (c) 3-D. weights), any local minimum point may not be found by optimizing a 2-1-1 MLP (five weights), since the hidden weights tend to be divergent (or weight-? attractors). Here is another example: Example 2: An N -shape curve fitting to four data: Data(x; t) ? {(?3; 0), (?1; 1), (1; 0), (3; 1)}. We solved ?E = 0 to find all stationary points of a two-weight 1-1-1 MLP with a logistic hiddennode function ?(x) ? 1+e1?x , and found p? ? 1.0185 and v? ? 0.3571 with k?E(? 0?)k = O(10?15), roughly the order of machine (double) precision, and E(? 0?) ? 0.4111. The Hessian ?2 E(? 0?) was positive definite (eigenvalues: 0.8254 and 1.4824). We also found a saddle point. There was another type of attractive points, where ? is driven to saturation due to a large hidden weight v in magnitude (weight-? attractors). Fig.2 displays those three types of stationary points. Clearly, for a rigorous proof of Conjecture, we need to characterize those different types, and clarify their underlying assumptions; yet, it is quite an arduous task because the situation totally depends on data; see also our Hessian argument for Blum?s line in Sec.3.2. We continue with Example 2 to verify the above theorems. We set ? = [?p? , (1 ? ?)p? , v? , v?] in a 1-2-1 MLP. When ? = 0.5, the Hessian ?2 E was positive semi-definite. If a small perturbation is added to v? , then ?2 E becomes indefinite (see Theorem 2). In contrast, when ? = ?1.5, ?2 E became indefinite (minimum eigenvalue ?0.2307); this situation was similar to Fig.1(left). Remarks: The eigen-spectrum (or curvature) variation along a line often arises in separable (i.e., mixed linear and nonlinear) optimization problems. As a small non-MLP model, consider, for instance, a separable objective function with ? ? [p, v]T , two variables alone: F (?) = F (p, v) = pv 2 . Expressed below are the gradient and ?Hessian ? ? of F : ? ?F =
v2 ; 2pv
?2 F =
0 2v
2v . 2p
(15)

Consider a line v = 0, where the Hessian ?2 F is singular. Then, the eigenspectrum of ?2 F changes as the linear parameter p alters while the first-order necessary condition (?F = 0) is maintained with the objective-function value F = 0 held fixed. Clearly, ?2 F is positive semi-definite when p ¿ 0, whereas it is negative semi-definite when p ¡ 0. Hence, the line is a collection of degenerate stationary points. In this way, singularities may be closely related to flat regions, where any updates of 5

parameters do not change the objective function value. Back to MLP-learning, Blum [10] describes a different linear manifold of stationary points (see Sec.3.2 for more details), where the ?-adjusting procedure described above fails because D = 0 (see Example 3 below also). Some other types of linear manifolds (and eigen-spectrum changes) can be found; e.g., in [11, 4, 3]; unlike their work, our paper did not claim anything about local minima, and our approach is totally different. Example 3: A linear-output five-weight 1-1-2-1 MLP with ? = [p1 , p2 —v1 , v2 —w1]T (no thresholds), having tanh-hidden-node functions. If ? ? = [1, 1, 0, 0, 0]T , then ?E(? ?) = 0 with the indefinite Hessian ?2 E (hence, ? ? a saddle point) below, in which all diagonal entries of S are zero due to D = 0: 2

?

? E(?) =

2

0 0 6 0 0 6 0 0 4 0 0 0 0

0 0 0 0 ?

0 0 0 0 ?

3 0 0 7 ? ? 7 5 ? 0

with ? ?

m X

xd r d .

d=1

Here, ? denotes a non-zero entry with input x and residual r over an arbitrary number m of data. 2 The point to note here is that it is important to look at the entire Hessian ?2 E of size n ? n. When H = O, a 4 ? 4 block of zeros, ?2 E would be indefinite (again by the interlace theorem) as long as non-zero off-diagonal entries exist in ?2 E , as in Example 3 above. Needless to say, however, the Hessian analysis fails in certain pathological cases (see Sec.3.2). Typical is an aforementioned weight-? case, where the sigmoid-shaped hidden-node functions are driven to saturation limits due to very large hidden weights. Then, only part of JT J associated with linear weights p appear in ?2 E since S = O even if residuals are still large. This case is outside the scope of our analysis. It should be noted that a regularization scheme to penalize large weights is quite orthogonal to our scheme to exploit negative curvature. If a regularization term ?? T ? (with non-negative scalar ?) is added to E, then the negative-curvature information will be lost due to ?2 E + ?I.

3

The 2-2-1 MLP-learning examples found in the literature

In this section, we consider learning with a 2-2-1 MLP having nine weights; then, Eq.(6) reduces to ? T ? [pT —vT] = [pT —v1T —v2T] = [p0 , p1 , p2 —v0,1 , v1,1 , v2,1 —v0,2 , v1,2 , v2,2], where vj is a (hidden) weight vector connecting to the jth hidden node. Here, all weights are nonlinear since both hidden and final outputs are produced by sigmoidal logistic function ?(x) ? 1+e1?x . 3.1

Insensitivity to the initial weights in the singular XOR problem

The world-renowned XOR problem (involving only four data of binary values: ON and off) with a standard nine-weight 2-2-1 MLP is inevitably a singular problem because the Gauss-Newton Hessian JT J in Eq.(4) is always singular (at most rank 4), whereas S tends to be of (nearly) full rank; so does ?2 E (cf. rank analysis in [12]). This implies that singularity in terms of JT J is everywhere in the posed neuro-manifolds. It is well-known (e.g., see [13]) that the origin (p = 0 and v = 0) is a singular saddle point, where ?E = 0 and ?2 E = JT J with only one positive eigenvalue and eight zeros. An interesting observation is that there always exists a descending path to the solution from any initial point ? init as long as ? init is randomly generated in a small range; i.e., in the vicinity of the origin. That is, first go directly down towards the origin from ? init , and then move in a descent direction of negative curvature so as to escape from that singular saddle point. In this way, the 2-2-1 MLP can develop insensitivity to initial weights, always solving the posed XOR problem. 3.2

Blum?s linear manifold of stationary points

In the XOR problem, Blum [10] found a line of stationary points by adding constraints to ? as L1 ? v0,1 = v0,2 , w1 ? v1,1 = v2,2 , w2 ? v1,2 = v2,1 , w ? p1 = p2 , (with L ? p0), (16) T leading to a weight-sharing MLP of five weights: ? ? [L, w, L1 , w1 , w2] following the notations in [10]. Using four XOR data: (x1 , x2 ; t) = {(0, 0; off), (0, 1; ON), (1, 0; ON), (1, 1; off)} for E in Eq.(3), Blum considered a point with v = 0; hence, ? ? ? [L, w, 0, 0, 0]T , which gives two identical 1 1 hidden-node outputs: h1 = h2 = ?(0) = 1+e 0 = 2 . This is the same situation as in Sec.2.2 and 2.3. By the constraints given in Eq.(16), the terminal output is given by y = ?(L + w). All those node outputs are independent of input data. Then, for a given target value ?off? (e.g., 0.1), set ON = 2?(L + w) ? off ?? ?(L + w) = (off + ON)/2 (17) so that those target values ?off? and ?ON? must approximate XOR. 6

Blum?s Claim (page 539 [10]): There are many stationary points that are not absolute minima. They correspond to w and L satisfying Eq.(17). Hence, they lie on a line ?L + w = c (constant)? in the (w, L)-plane. Actually, these points are local minima of E, being 21 (ON ? off)2 . 2 A little algebra confirms that ?E = 0, and the quantities corresponding to e and D in Eq.(11) are all zeros; hence, S = O. Consequently, no matter how ? (see Theorem 5) is changed to update w and L (along the line), ?2 E stays positive semi-definite, and E in Eq.(3) remains the same 0.5 (flat region). This is certainly a limitation of the second-order Hessian analysis, and thus more efforts using higher-order derivatives were needed to disprove Blum?s claim (see [14, 15]), and it turned out that Blum?s line is a collection of singular saddle points. In what follows, we show what conditions must hold for the Hessian argument to work. The 5-by-5 Hessian matrix ?2 E at a stationary point ? ? = [L, w, 0, 0, 0] is given by 2

4A 4A 2wA 4A 2wA 6 4A 6 2 6 2wA 2wA w 2 A ? E = — {z } 6 4 wA wA w2 A 5?5 2 2 wA wA w2 A

wA wA w2 A 2 w2 (3A+S) 8 w2 (3A+S) 8

3 wA 8 2 0 wA ¿ 7 ¡A ? {? (L + w)} 2 7 w 7 A ? ? 2 7 with ¿ w2 :S ? ?00

(L+w) ?(L+w)? off . 5 (3A+S) 8 w2 (3A+S) 8

(18)

2

We thus obtain two non-zero eigenvalues of ?2 E , ?1 and ?2 , below using k ? w8 (3A + S): ?1 , ?2 =

1 2

?

?

ff ? q A(w2 + 8) + 2k ? [A(w 2 + 8) + 2k]2 ? 2A(w 2 + 8)(4k ? w 2 A) .

(19)

Now, the smaller eigenvalue can be rendered negative when the following condition holds: ? ? 2 4k ? w 2 A ¡ 0 ?? A + S = {?0 (L + w)} + ?00 (L + w) ?(L + w) ? off ¡ 0.

(20)

Choosing L+w = 2 and off = 0.1 accomplishes our goal, yielding sign-different eigenvalues with ON = 2?(2)?off ? 1.6616 by Eq.(17). Because ?2 E is indefinite, the posed stationary point is a saddle point with E = 12 (ON ? off)2 (? 1.219), as desired. In other words, the target value for ON is modified to break symmetry in data. Such a large target value ON (as 1.6616) is certainly unattainable outside the range (0,1) of the sigmoidal logistic function ?(x), but notice that ON is often set equal to 1.0, which is also un-attainable for finite weight values. It appears that the choice of such a (fictitiously large) value ON does not violate any Blum?s assumption. When 0 ? off ¡ ON ? 1 (with w 6= 0), the Hessian ?2 E in Eq.(18) is always positive semi-definite of rank 2. Hence, it is a singular saddle point. 3.3

Two-class pattern classification problems of Gori and Tesi

We next consider two two-class pattern classification problems made by Gori & Tesi: one with five binary data (p.80 in [17]), and another with only three data (p.93 in [16]); see Fig.3. Both are singular problems, because rank(JT J) ? 5; yet, both S and ?2 E tend to be of full rank; therefore, the 9?9 Hessian ?2 E tends to be indefinite (see Theorems 1 and 2). On p.81 in [17], a configuration of two separation lines, like two solid lines given by ? init in Fig.3(left) and (right), is claimed as a region attractive to a local-minimum point. Indeed, the batch-mode steepest-descent method fails to change the orientation of those solid lines. But its failure does not imply that there is no descending way out of the two-solid-line configuration given by ? init because the convergence of the steepestdescent method to a (local) minimizer can be guaranteed by examining negative curvature (e.g., p.45 in [18]). We shall show a descending negative curvature direction. In the five-data case, the steepest-descent method moves ? init to a point, where the weights become relatively large; the gradient vector ?E ? 0; the Hessian ?2 E is positive semi-definite; and Eq.(3) with m = 5 is given by E = 31 (ON ? off)2 , for which the two residuals at data points (0,0) and (1,1) are made zeros. We can find such a point analytically by a linear-equation solving: Given ? init in Fig.3, the solution to the linear system below yields p? = [p?0 , p?1 , p?2]T (three terminal weights): 2

1 41 1

?(?1.5) ?(0.5) ?(?0.5)

3 32 ? 3 2 ??1 (off) ?(?0.5) p0 ?1 ?(1.5) 54 p?1 5 = 4 ?? (off) ? 5 . ??1 2 ON3+off p?2 ?(0.5)

The resulting point ? ? ? [p?0 , p?1 , p?2 ; ?1.5, 1, 1; ?0.5, 1, 1]T, where the norm of p? becomes relatively large O(102), gives the zero gradient vector, the positive semi-definite Hessian of rank 5, and E = 13 (ON ? off)2 , as mentioned above. It is observed, however, that small perturbations on ? ? render 7

net = x 1 + x 2 ? 1.5

x2

net = ? x 1 + x 2 ? 0.5

x2 0

1.5

1.5

?1

1

(0, 1)

(0, 1)

(1, 1)

h2

h1 0

0.5

1

0.5

1.5

1

1.5 0.5 ?0.5

(1, 0)

x1

?1.5

1

1

(0, 0)

0

1

(1, 0)

x1

?0.5

1 ?0.5

0.5

1

2

x1

x2

net = ? x1 + x2 + 0.5

net = x 1 + x 2 ? 0.5

Figure 3: Gori & Tesi?s two-class pattern classification problems (left) three-data case; (right) fivedata case; and (middle) a 2-2-1 MLP with initial weight values ? init ? [0, 1, ?1; ?1.5, 1, 1; ?0.5, 1, 1]T . Its corresponding initial configuration gives two solid lines of net-inputs (to two hidden nodes) in the input space, where ??? stands for two ON-data (1,0), (0,1), whereas ??? for one off-data (0.5,0.5) in left figure and three off-data (0,0), (0.5,0.5), (1,1) in right figure. A solution to both problems may be given by the two dotted lines with ? sol ? [0, 1, ?1; ?0.5, ?1, 1; 0.5, ?1, 1]T . ?2 E indefinite of full rank (since S is dominant): rank(S) = rank(?2 E) = 9 with rank(JT J) = 4; this

suggests a descend direction (other than the steepest descent) to follow from ? init to a solution ? sol . Fig.3(right) presents one of them, an intuitive change of six hidden weights (with the other three weights held fixed) from two solid lines to two dotted ones, indicated by two thick arrows given by ?? ? ? sol ? ? init = [0, 0, 0; 1, ?2, 0; 1, ?2, 0]T, is a descent direction of negative curvature down to ? sol because ?? T ?2E(? init)?? ¡ 0, where ?2E(? init), the Hessian evaluated at ? init , was indefinite. Intriguingly enough, it is easy to confirm for the three-data case that the posed ?descent? direction of negative curvature ?? is orthogonal to ??E, the steepest-descent direction. Claim: Line search from ? init to ? sol monotonically decreases the squared error E (? init + ???) as the step size ? (scalar) changes from 0 to 1; hence, no plateau. Proof for the three-data case: (The five-data case can be proved in a similar fashion.) Using target values ON=1 and off=0, let q(?) ? E(? init +???)?E(? init). Then, we show below that q 0 (?) ¡ 0 using a property that ?(?x) = 1??(x): q(?) = 12 {?(?0.5??)??(0.5??)?ON}2 + 12 {?(?0.5+?)??(0.5+?)?ON}2 ?{?(?0.5)??(0.5)?ON}2 = 12 {1??(0.5+?)??(0.5??)?ON}2 + 12 {1??(0.5??)??(0.5+?)?ON}2?{1??(0.5)??(0.5 = {1?ON??(0.5 + ?)??(0.5 ? ?)}2 ? {1 ? ON ? 2?(0.5)}2 = {?(0.5 + ?) + ?(0.5 ? ?)}2 ? 4 {?(0.5)}2 .

Differentiation leads to q 0 (?) = 2 {?(0.5+?)+?(0.5??)} {?0 (0.5+?)??0 (0.5??)} ¡ 0 because ?(0.5+?) ¿ 0, ?(0.5??) ¿ 0, and ? ¿ 0, which guarantees ?0 (0.5+?) ¡ ?0 (0.5??). 2

4 Summary

In a general setting, we have proved that negative curvature can arise in MLP-learning. To make it analytically tractable, we intentionally used noise-free small data sets but on ?noisy? data, the conditions for Theorems 1 and 2 most likely hold in the vicinity of singularity regions; it then follows that the Hessian ?2 E tends to be indefinite (of nearly full rank). Our numerical results confirm that the negative-curvature information is of immense value for escaping from singularity plateaus including some problems where no method was developed to alleviate plateaus. In simulation, we employed the second-order stagewise backpropagation [12] (that can evaluate ? 2 E and JT J at the essentially same cost; see proof therein) to obtain ?2 E explicitly and its eigen-directions so as to exploit negative curvature. This approach is suitable for up to medium-scale problems, for which our analysis suggests using existing trust-region globalization strategies whose theory has thrived on negative curvature including indefinite dogleg [19]. For large-scale problems, one could resort to matrix-free Krylov subspace methods: Among them, the truncated conjugate-gradient (Krylovdogleg) method tends to pick up an arbitrary negative curvature (hence, slowing down learning; see [20] for numerical evidence); so, other trust-region Krylov subspace methods are of our great interest such as a Lanczos type [21] and a parameterized eigenvalue approach [22]. Acknowledgments The work is partially supported by the National Science Council, Taiwan (NSC-99-2221-E-011-097). 8

2 References

[1] Amari, S.-I., Park,H. & Fukumizu, K. Adaptive Method of Realizing Natural Gradient Learning for Multilayer Perceptrons. Neural Computation, 12:1399-1409, 2000. [2] Amari, S.-I., Park, H. & Ozeki, T. Singularities affect dynamics of learning in neuro-manifolds. Neural Computation, 18(5):1007-1065, 2006. [3] Wei, H., Zhang, J., Cousseau, F., Ozeki, T., & Amari, S.-I. Dynamics of Learning Near Singularities in Layered Networks. Neural Computation, 20(3):813-843, 2008. [4] Fukumizu, K. & Amari, S.-I. Local Minima and Plateaus in Hierarchical Structures of Multilayer Perceptrons. Neural Networks, 13(3):317?327, 2000. [5] Jennrich, R.I. & Sampson, P.F. Application of Stepwise Regression to Non-Linear Estimation. Technometrics, 10(1):63?72, 1968. [6] Cousseau, F., Ozeki, T., & Amari, S.-I. Dynamics of Learning in Multilayer Perceptrons near Singularities IEEE Trans. on Neural Networks, 19(8):1313-1328, 2008. [7] Powell, M.J.D. A hybrid method for nonlinear equations. In Numerical Methods for Nonlinear Algebraic Equations, Ed. by P.Rabinowitz, Gordon & Breach, London, pp.87?114, 1970. [8] Dennis, J.E., Jr. Nonlinear least squares and equations. In The state of the art in numerical analysis, Ed. by D. Jacobs, Academic Press, London, pp.269?312, 1977. [9] Parlett, B.N. The Symmetric Eigenvalue Problem. SIAM, 1998. [10] Blum, E.K. Approximation of Boolean Functions by Sigmoidal Networks: Part I: XOR and other twovariable functions. Neural Computation, 1:532-540, 1989. [11] Sprinkhuizen-Kuyper, I.G. & Boers, E.J.W. A Local Minimum for the 2-3-1 XOR Network. IEEE Transactions on Neural Networks, 10(4):968?971, 1999. [12] Mizutani, E. & Dreyfus, S.E. Second-order stagewise backpropagation for Hessian-matrix analyses and investigation of negative curvature. Neural Networks, vol.21 (issues 2?3):193-203, 2008. (See its Corrigendum in vol.21, issue 9, page 1418). [13] Sprinkhuizen-Kuyper, I.G. & Boers, E.J.W. The error surface of the 2-2-1 XOR network: The finite stationary points. Neural Networks, 11:683?690, 1998. [14] Tsaih, R.-H. An Improved Back Propagation Neural Network Learning Algorithm. Ph.D thesis at the Department of Industrial Engineering and Operations Research, University of California at Berkeley, pp.67?70, 1991. [15] Sprinkhuizen-Kuyper, I.G. & Boers, E.J.W. A comment on a paper of Blum: Blum?s local minima are saddle points. Tech. Rep. No. 94-34, Leiden University, Department of Computer Science, Leiden, The Netherlands, 1994. [16] Gori, M. & Tesi, A. Some examples of local minima during learning with backpropagation. Third Italian Workshop on Parallel Architectures and Neural Networks. (Ed. by E.R. Caianiello), World

Scientific Publishing Co., pp. 87?94, 1990. [17] Gori, M. & Tesi, A. On the Problem of Local Minima in Backpropagation. IEEE Trans. on Pattern Analysis and Machine Intelligence, 14(1):76-86, 1992. [18] Nocedal, J & Wright, S.J. Numerical Optimization. Springer Verlag, 1999. [19] Byrd, R.H., Schnabel, R.B. & Schultz, G.A. Approximate solution of the trust region problems by minimization over two-dimensional subspaces. Mathematical Programming, 40:247?263, 1988. [20] Mizutani, E. & Demmel, J.W. Iterative scaled trust-region learning in Krylov subspaces via Pearlmutter?s implicit sparse Hessian-vector multiply. In S. Thrun, L. Saul, and B. Sch o? lkopf, editors, Advances in Neural Information Processing Systems, MIT Press, 16:209?216, 2004. [21] Gould, N.I.M., Lucidi, S., Roma, M. & Toint, Ph.L. Solving the trust-region subproblem using the Lanczos method. SIAM Journal on Optimization, 9(2):504?525, 1999. [22] Rojas, M., Santos, S.A. & Sorensen, D.C. A New Matrix-Free Algorithm for the Large-Scale TrustRegion Subproblem. SIAM Journal on Optimization, 11(3):611?646, 2000.

9