

1 Stato dell'arte

1.1 Andy M. Sarroff, Dartmouth college

1.2 Nils Mönning & Suresh Manandhar, University of York

Nils Mönning e Suresh Manandhar hanno comparato diversi MLP (multi layer perceptron) complessi con dei corrispettivi MLP reali. Per evitare casistiche particolari sono stati utilizzati 4 database differenti (MNIST digit classification, CIFAR-10 image classification, CIFAR-100 image classification, Reuters topic classification) e per ridurre l'impatto delle inizializzazioni, ogni modello ha effettuato 10 training. É tenuto fisso il numero di elementi per layer (64 neuroni nel caso reale e 32 in quello complesso) e per il calcolo del gradiente nel caso complesso è stato utilizzato Wirtinger. I risultati ottenuti sono mostrati nelle tabelle seguenti:

Tabella 1) Test di MLP con $k + 2$ layers, ognuno di 64 elementi (64 neuroni per la rete reale e 32 per quella complessa) e output di 10 elementi.

Hidden layers k	Real parameters $p_{\mathbb{R}}$	Activation function φ	MNIST	
			\mathbb{R}	\mathbb{C}
$k = 0$	50,816	<i>identity</i>	0.9282	0.9509
		<i>tanh</i>	0.9761	0.9551
		<i>relu</i>	0.9780	0.9710
		$ z ^2$	0.9789	0.9609
		$ z $	0.9770	0.9746
$k = 2$	59,008	<i>identity</i>	0.9274	0.9482
		<i>tanh</i>	0.9795	0.8923
		<i>relu</i>	0.9804	0.9742
		$ z ^2$	0.9713	0.6573
		$ z $	0.9804	0.9755
$k = 4$	67,200	<i>identity</i>	0.9509	0.9468
		<i>tanh</i>	0.9802	0.2112
		<i>relu</i>	0.9816	0.9768
		$ z ^2$	0.8600	0.2572
		$ z $	0.9789	0.9738
$k = 8$	83,584	<i>identity</i>	0.9242	0.1771
		<i>tanh</i>	0.9796	0.1596
		<i>relu</i>	0.9798	0.9760
		$ z ^2$	0.0980	0.0980
		$ z $	0.9794	0.1032

Tabella 2) Test di MLP con $k + 2$ layers, ognuno di 64 elementi (64 neuroni per la rete reale e 32 per quella complessa) e output di 46 elementi.

Hidden layers k	Real parameters $p_{\mathbb{R}}$	Activation function φ	Reuters	
			\mathbb{R}	\mathbb{C}
$k = 0$	642,944	<i>identity</i>	0.8116	0.7939
		<i>tanh</i>	0.8117	0.7912
		<i>relu</i>	0.8081	0.7934
		$ z ^2$	0.8050	0.7885
		$ z $	0.8068	0.7992
$k = 2$	651,136	<i>identity</i>	0.8005	0.7836
		<i>tanh</i>	0.7978	0.7320
		<i>relu</i>	0.7921	0.7854
		$ z ^2$	0.7725	0.6874
		$ z $	0.7996	0.7823
$k = 4$	659,328	<i>identity</i>	0.7925	0.7787
		<i>tanh</i>	0.7814	0.4199
		<i>relu</i>	0.7734	0.7671
		$ z ^2$	0.5895	0.0650
		$ z $	0.7863	0.7694
$k = 8$	675,712	<i>identity</i>	0.7929	0.7796
		<i>tanh</i>	0.7542	0.1861
		<i>relu</i>	0.7555	0.7676
		$ z ^2$	0.0053	0.0053
		$ z $	0.7671	0.7524

Tabella 3) Test di MLP con $k + 2$ layers, ognuno di 128 elementi (128 neuroni per la rete reale e 64 per quella complessa) e output di 10 elementi.

Hidden layers k	Real parameters $p_{\mathbb{R}}$	Activation function φ	CIFAR-10	
			\mathbb{R}	\mathbb{C}
$k = 0$	394,496	<i>identity</i>	0.4044	0.1063
		<i>tanh</i>	0.4885	0.1431
		<i>relu</i>	0.4902	0.4408
		$ z ^2$	0.5206	0.1000
		$ z $	0.5256	0.1720
$k = 2$	427,264	<i>identity</i>	0.4039	0.1000
		<i>tanh</i>	0.5049	0.1672
		<i>relu</i>	0.5188	0.496
		$ z ^2$	0.1451	0.1361
		$ z $	0.5294	0.1000
$k = 4$	460,032	<i>identity</i>	0.4049	0.1000
		<i>tanh</i>	0.4983	0.1549
		<i>relu</i>	0.8445	0.6810
		$ z ^2$	0.1000	0.1000
		$ z $	0.5273	0.1000
$k = 8$	525,568	<i>identity</i>	0.4005	0.1027
		<i>tanh</i>	0.4943	0.1365
		<i>relu</i>	0.5072	0.4939
		$ z ^2$	0.1000	0.1000
		$ z $	0.5276	0.1000

Tabella 4) Test di MLP con $k + 2$ layers, ognuno di 128 elementi (128 neuroni per la rete reale e 64 per quella complessa) e output di 100 elementi.

Hidden layers k	Real parameters $p_{\mathbb{R}}$	Activation function φ	CIFAR-100	
			\mathbb{R}	\mathbb{C}
$k = 0$	406,016	<i>identity</i>	0.1758	0.0182
		<i>tanh</i>	0.2174	0.0142
		<i>relu</i>	0.1973	0.1793
		$ z ^2$	0.2314	0.0158
		$ z $	0.2423	0.0235
$k = 2$	438,784	<i>identity</i>	0.1720	0.0100
		<i>tanh</i>	0.2314	0.0146
		<i>relu</i>	0.2400	0.2123
		$ z ^2$	0.0143	0.0123
		$ z $	0.2411	0.0100
$k = 4$	471,552	<i>identity</i>	0.1685	0.0100
		<i>tanh</i>	0.2178	0.0157
		<i>relu</i>	0.2283	0.2059
		$ z ^2$	0.0109	0.0100
		$ z $	0.2313	0.0100
$k = 8$	537,088	<i>identity</i>	0.1677	0.0100
		<i>tanh</i>	0.2000	0.0130
		<i>relu</i>	0.2111	0.1956
		$ z ^2$	0.0100	0.0100
		$ z $	0.2223	0.0100

Tabella 5) Test di MLP con $k + 2$ layers, 500000 parametri reali e output di 10 elementi.

Hidden layers k	Units		Activation function φ	CIFAR-10	
	$m_{\mathbb{R}}$	$m_{\mathbb{C}}$		\mathbb{R}	\mathbb{C}
$k = 0$	630	315	<i>identity</i>	0.9269	0.9464
			<i>tanh</i>	0.9843	0.9467
			<i>relu</i>	0.9846	0.9828
			$ z ^2$	0.9843	0.9654
			$ z $	0.9857	0.9780
$k = 2$	339	207	<i>identity</i>	0.9261	0.9427
			<i>tanh</i>	0.9852	0.6608
			<i>relu</i>	0.9878	0.9835
			$ z ^2$	0.9738	0.8331
			$ z $	0.9852	0.9748
$k = 4$	268	170	<i>identity</i>	0.9254	0.2943
			<i>tanh</i>	0.9838	0.2002
			<i>relu</i>	0.9862	0.9825
			$ z ^2$	0.8895	0.2875
			$ z $	0.9846	0.9870
$k = 8$	205	134	<i>identity</i>	0.9250	0.1136
			<i>tanh</i>	0.9810	0.1682
			<i>relu</i>	0.9851	0.9824
			$ z ^2$	0.0980	0.0980
			$ z $	0.9803	0.1135

Tabella 6) Test di MLP con $k + 2$ layers, 500000 parametri reali e output di 46 elementi.

Hidden layers k	Units		Activation function φ	Reuters	
	$m_{\mathbb{R}}$	$m_{\mathbb{C}}$		\mathbb{R}	\mathbb{C}
$k = 0$	50	25	<i>identity</i>	0.8072	0.7970
			<i>tanh</i>	0.8112	0.7832
			<i>relu</i>	0.8054	0.7925
			$ z ^2$	0.8037	0.7929
			$ z $	0.8059	0.7912
$k = 2$	49	25	<i>identity</i>	0.7992	0.7809
			<i>tanh</i>	0.7952	0.7289
			<i>relu</i>	0.7898	0.7751
			$ z ^2$	0.7778	0.6887
			$ z $	0.7716	0.7911
$k = 4$	49	25	<i>identity</i>	0.7636	0.7854
			<i>tanh</i>	0.7796	0.4550
			<i>relu</i>	0.7658	0.7676
			$ z ^2$	0.5823	0.0289
			$ z $	0.7809	0.7573
$k = 8$	48	24	<i>identity</i>	0.7760	0.7663
			<i>tanh</i>	0.7449	0.1799
			<i>relu</i>	0.7182	0.7484
			$ z ^2$	0.0053	0.0053
			$ z $	0.7449	0.7302

Tabella 7) Test di MLP con $k + 2$ layers, 500000 parametri reali e output di 10 elementi.

Hidden layers k	Units		Activation function φ	MNIST	
	$m_{\mathbb{R}}$	$m_{\mathbb{C}}$		\mathbb{R}	\mathbb{C}
$k = 0$	162	81	<i>identity</i>	0.4335	0.1006
			<i>tanh</i>	0.5032	0.1676
			<i>relu</i>	0.5007	0.4554
			$ z ^2$	0.5179	0.1006
			$ z $	0.5263	0.2381
$k = 2$	148	77	<i>identity</i>	0.4069	0.1000
			<i>tanh</i>	0.5205	0.1673
			<i>relu</i>	0.5269	0.4963
			$ z ^2$	0.1395	0.1273
			$ z $	0.5315	0.1000
$k = 4$	138	74	<i>identity</i>	0.4052	0.1000
			<i>tanh</i>	0.5218	0.1475
			<i>relu</i>	0.5203	0.4975
			$ z ^2$	0.1065	0.1010
			$ z $	0.5234	0.1000
$k = 8$	123	69	<i>identity</i>	0.4050	0.1003
			<i>tanh</i>	0.5162	0.1396
			<i>relu</i>	0.5088	0.4926
			$ z ^2$	0.1000	0.1000
			$ z $	0.5194	0.1000

Tabella 8) Test di MLP con $k + 2$ layers, 500000 parametri reali e output di 100 elementi.

Hidden layers k	Units		Activation function φ	CIFAR-100	
	$m_{\mathbb{R}}$	$m_{\mathbb{C}}$		\mathbb{R}	\mathbb{C}
$k = 0$	158	79	<i>identity</i>	0.2807	0.0314
			<i>tanh</i>	0.2308	0.0193
			<i>relu</i>	0.2153	0.1935
			$ z ^2$	0.2364	0.0124
			$ z $	0.2439	0.0279
$k = 2$	144	75	<i>identity</i>	0.1723	0.0100
			<i>tanh</i>	0.2440	0.0203
			<i>relu</i>	0.2481	0.2224
			$ z ^2$	0.0155	0.0151
			$ z $	0.2453	0.0100
$k = 4$	135	72	<i>identity</i>	0.1727	0.0100
			<i>tanh</i>	0.2397	0.0150
			<i>relu</i>	0.2381	0.2147
			$ z ^2$	0.0122	0.0100
			$ z $	0.2390	0.0100
$k = 8$	121	67	<i>identity</i>	0.1706	0.0100
			<i>tanh</i>	0.2209	0.0164
			<i>relu</i>	0.2167	0.2027
			$ z ^2$	0.0100	0.0100
			$ z $	0.2191	0.0100

1.3 Nitzan Guberman

Viene costruita una CNN (convolutional neural network) complessa per determinare se una determinata patch di immagine contiene una cellula e confrontare questa rete con la sua controparte reale. Le due reti mostrano risultati comparabili, sebbene la rete complessa soffra di difficoltà di convergenza. Per verificare l'affermazione che le CNN complesse agiscano come regolarizzatori, esaminiamo il comportamento della *loss function*. La CNN con valore reale è significativamente più vulnerabile all'overfitting. Confrontiamo la rete complessa con il suo equivalente reale. Questa rete condivide la stessa architettura di quella complessa, solo con il doppio dei canali e dei kernel di convoluzione. Per costruzione, l'ultimo strato della rete reale è costituito da un numero doppio di neuroni rispetto a quello complesso. Poiché le etichette sono binarie, il livello finale deve essere

a due canali, quindi aggiungiamo un affine layer completamente connesso che sostituisce il livello di proiezione nella rete complessa.

