IA353A - Neural Networks
EC1

Rafael Claro Ito
(R.A.: 118430)

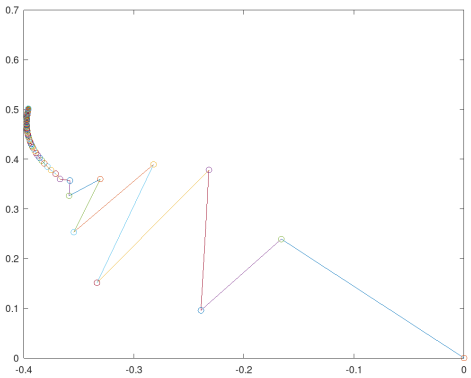
May 2020

Question 1

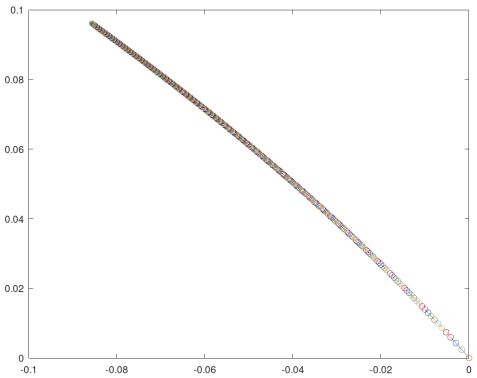
1.1 SGD

1.1.1 Underdetermined

```
% SGD para o caso de sistema linear subdeterminado
clear all;
randn('state',0);
N = 10;
Nit = 500;
X = randn(N,2*N);
S = sign(randn(N,1));
w = (X'/(X*X'))*S;
w1 = zeros(2*N,1);
%passo = 0.1;
passo = 0.001;
for it=2:Nit,
    w1(:,it) = w1(:,it-1) - (passo/sqrt(it))*(X'*X*w1(:,it-1)-X'*S);
end
figure(1);
title('Stochastic Gradient Descent');
for it = 1:(Nit-1),
    plot([w1(1,it);w1(1,it+1)],[w1(2,it);w1(2,it+1)]);hold on;
    plot(w1(1,it),w1(2,it),'o');
    plot(w1(1,it+1),w1(2,it+1),'o');
end
hold off;
disp('[Minimum Norm Solution Obtained solution]');
disp([w w1(:,Nit)]);
[S X*w X*w1(:,Nit)]
%-----------------------------
% save figure
path = strcat('../figures/Q1/sgd_under_step_', num2str(passo), '.png');
saveas(gcf, path);
```



(a) Progression of W using step of 0.1

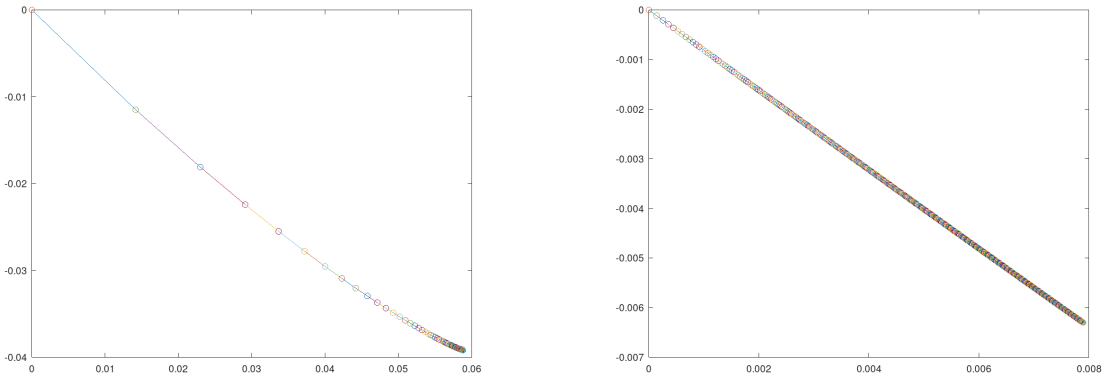


(b) Progression of W using step of 0.001

Figure 2: Progression of W for underdetermined system using SGD

1.1.2 Overdetermined

```
% SGD para o caso de sistema linear sobredeterminado
clear all;
randn('state',0);
N = 10;
Nit = 500;
X = randn(N,2);
S = sign(randn(N,1));
w = (X'*X)\X'*S;
disp('Optimal solution');
disp(w);
w1 = zeros(2,1);
%passo = 0.1;
passo = 0.001;
for it=2:Nit,
    w1(:,it) = w1(:,it-1) - (passo/sqrt(it))*(X'*X*w1(:,it-1)-X'*S);
end
figure(1);
title('Stochastic Gradient Descent');
for it = 1:(Nit-1),
    plot([w1(1,it);w1(1,it+1)],[w1(2,it);w1(2,it+1)]);hold on;
    plot(w1(1,it),w1(2,it),'o');
    plot(w1(1,it+1),w1(2,it+1),'o');
end
hold off;
disp('Obtained solution');
disp(w1(:,Nit));
[S X*w X*w1(:,Nit)]
%------------------------------
% save figure
path = strcat('../figures/Q1/sgd_over_step_', num2str(passo), '.png');
saveas(gcf, path);
```



(a) Progression of W using step of 0.1

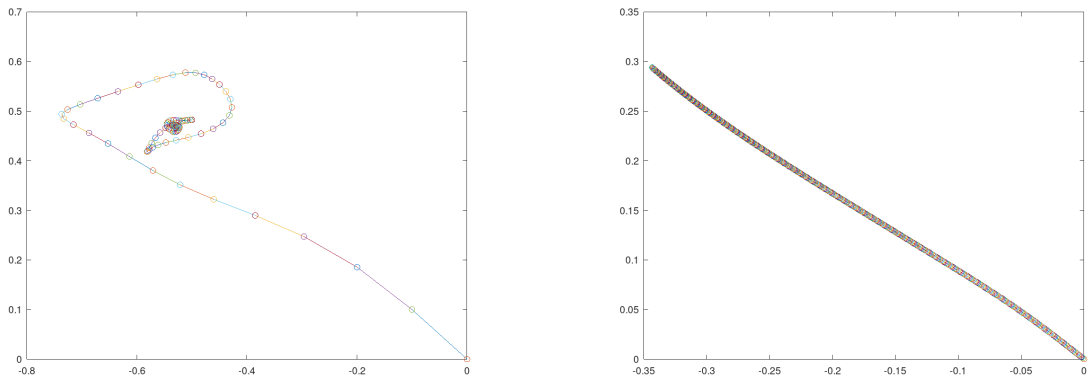
(b) Progression of W using step of 0.001

Figure 3: Progression of W for overdetermined system using SGD

1.2 Adam

1.2.1 Underdetermined

```
% Adam para o caso de sistema linear subdeterminado
clear all;
randn('state',0);
N = 10;
Nit = 500;
X = randn(N,2*N);
S = sign(randn(N,1));
w = (X'/(X*X'))*S;      % optimal solution
w1 = zeros(2*N,1);      % initial weights
%passo = 0.1;
passo = 0.001
%-----------------------------
% parameters
beta_1 = 0.9;
beta_2 = 0.999;
e = 1e-8;
m = v = m_hat = v_hat = zeros(2*N,1);
%-----------------------------
% loop
for it=2:Nit,
    g = X'*X*w1(:,it-1)-X'*S;
    m = beta_1*m + (1-beta_1)*g;
    v = beta_2*v + (1-beta_2)*g.^2;
    m_hat = m / (1 - beta_1^(it-1));
    v_hat = v / (1 - beta_2^(it-1));
    w1(:,it) = w1(:,it-1) - passo./(sqrt(v_hat) + e) .* m_hat;
    %w1(:,it) = w1(:,it-1) - (passo/sqrt(it))*(X'*X*w1(:,it-1)-X'*S);
end
%-----------------------------
% plot weights
figure(1);
title('Stochastic Gradient Descent');
for it = 1:(Nit-1),
    plot([w1(1,it);w1(1,it+1)],[w1(2,it);w1(2,it+1)]);hold on;
    plot(w1(1,it),w1(2,it),'o');
    plot(w1(1,it+1),w1(2,it+1),'o');
end
hold off;
disp('[Minimum Norm Solution Obtained solution]');
disp([w w1(:,Nit)]);
[S X*w X*w1(:,Nit)]
%-----------------------------
% save figure
path = strcat('../figures/Q1/adam_under_step_', num2str(passo), '.png');
saveas(gcf, path);
```



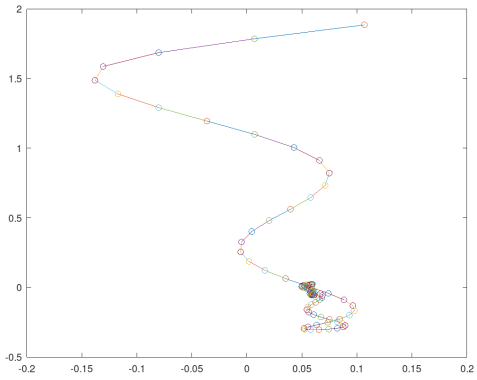
(a) Progression of W using step of 0.1

(b) Progression of W using step of 0.001

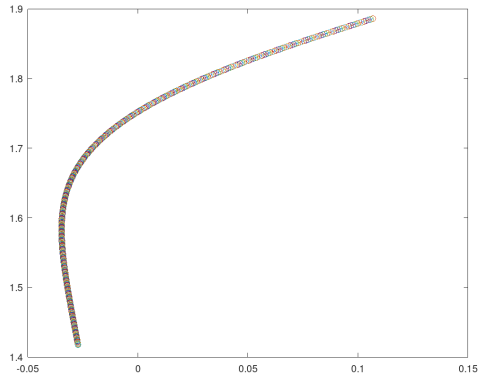
Figure 4: Progression of W for underdetermined system using Adam

1.2.2 Overdetermined

```
% Adam para o caso de sistema linear sobredeterminado
clear all;
randn('state',0);
N = 10;
Nit = 500;
X = randn(N,2);
S = sign(randn(N,1));
w = (X'*X)\X'*S;              % optimal solution
disp('Optimal solution');
disp(w);
w1 = randn(2,1);
passo = 0.1;
%passo = 0.001
%-----------------------------
% parameters
beta_1 = 0.9;
beta_2 = 0.999;
e = 1e-8;
m = v = m_hat = v_hat = zeros(2,1);
%-----------------------------
% loop
for it=2:Nit,
    g = X'*X*w1(:,it-1)-X'*S;
    m = beta_1*m + (1-beta_1).*g;
    v = beta_2*v + (1-beta_2).*g.^2;
    m_hat = m / (1 - beta_1^(it-1));
    v_hat = v / (1 - beta_2^(it-1));
    w1(:,it) = w1(:,it-1) - passo./(sqrt(v_hat) + e) .* m_hat;
    %w1(:,it) = w1(:,it-1) - (passo/sqrt(it))*(X'*X*w1(:,it-1)-X'*S);
end
%-----------------------------
% plot weights
figure(1);
title('Stochastic Gradient Descent');
for it = 1:(Nit-1),
    plot([w1(1,it);w1(1,it+1)],[w1(2,it);w1(2,it+1)]);hold on;
    plot(w1(1,it),w1(2,it),'o');
    plot(w1(1,it+1),w1(2,it+1),'o');
end
hold off;
disp('Obtained solution');
disp(w1(:,Nit));
[S X*w X*w1(:,Nit)]
%-----------------------------
% save figure
path = strcat('../figures/Q1/adam_over_step_', num2str(passo), '.png');
saveas(gcf, path);
```



(a) Progression of W using step of 0.1



(b) Progression of W using step of 0.001

Figure 5: Progression of W for overdetermined system using Adam

1.3 Comparison

	λ optimum	
	MSE	Accuracy
coarse search	64	1024
fine search	51.5	1091.8

Table 1: Values of regularization coefficient found in coarse and fine searches

Question 2

$$min\|Ax-b\|_P^2 + \|x-x_0\|_Q^2$$

Since we are searching for x that minimizes the previous expression, we will calculate the derivative with relation to x and set it equal to zero:

$$\frac{d}{dx}(\|Ax-b\|_P^2 + \|x-x_0\|_Q^2) = 0$$

Property used: $\|x\|_Q^2 = x^TQx$

$$\frac{d}{dx}[\overbrace{(Ax-b)^TP(Ax-b)}^{\|Ax-b\|_P^2} + \overbrace{(x-x_0)^TQ(x-x_0)}^{\|x-x_0\|_Q^2}] = 0$$

Property used: $(M+N)^T = M^T + N^T$

$$\frac{d}{dx}\{[(Ax)^T - b^T]P(Ax-b) + (x^T - x_0^T)Q(x-x_0)\} = 0$$

Property used: $(MN)^T = N^TM^T$

$$\frac{d}{dx}\{[x^TA^T - b^T]P(Ax-b) + (x^T - x_0^T)Q(x-x_0)\} = 0$$

$$\frac{d}{dx}[(x^TA^TPAx - x^TA^TPb - b^TPAx + b^TPb) + (x^TQx - x^TQx_0 - x_0^TQx + x_0^TQx_0)] = 0$$

Properties used:

- $\frac{d}{dy}(y^TMy) = M^Ty + My$.
- $\frac{d}{dy}(y^TMy) = 2My$, if $M = M^T$ (i.e. M is symmetric)
- $\frac{d(Ax)}{dx} = A$
- $\frac{d(x^TA)}{dx} = A^T$
- obs.: A^TPA is symmetric, since $(A^TPA)^T = (PA)^T(A^T)^T = A^TP^TA$, but $P^T = P$, since P is symmetric. Then $A^TPA = (A^TPA)^T$

Using the previous properties, we have:

$$\frac{d}{dx}[(x^TA^TPAx - x^TA^TPb - b^TPAx + b^TPb) + (x^TQx - x^TQx_0 - x_0^TQx + x_0^TQx_0)] = 0$$

$$[2A^TPAx - (A^TPb)^T - b^TPA + 0] + [2Qx - (Qx_0)^T - x_0^TQ + 0] = 0$$
$$2A^TPAx - (Pb)^T(A^T)^T - b^TPA + 2Qx - x_0^TQ^T - x_0^TQ = 0$$
$$2A^TPAx - b^TP^TA - b^TPA + 2Qx - x_0^TQ - x_0^TQ = 0$$
$$2A^TPAx - b^TPA - b^TPA + 2Qx - x_0^TQ - x_0^TQ = 0$$
$$2A^TPAx - 2b^TPA + 2Qx - 2x_0^TQ = 0$$
$$2A^TPAx + 2Qx = 2b^TPA + 2x_0^TQ$$
$$A^TPAx + Qx = b^TPA + x_0^TQ$$
$$(A^TPA + Q)^{-1}(A^TPA + Q)x = (A^TPA + Q)^{-1}(b^TPA + x_0^TQ)$$
$$x = (A^TPA + Q)^{-1}(b^TPA + x_0^TQ)$$
$$x = (A^TPA + Q)^{-1}(b^TP^TA + x_0^TQ^T)$$
$$x = (A^TPA + Q)^{-1}[(Pb)^TA + (Qx_0)^T]$$
$$x = (A^TPA + Q)^{-1}[(A^TPb)^T + (Qx_0)^T]$$
$$\boxed{x = (A^TPA + Q)^{-1}(A^TPb + Qx_0)^T}$$

Question 3

a)

If M is symmetric, then M^T must be equal to M:

$$M^T = (N^T N)^T = N^T (N^T)^T = N^T N = M$$

Now we need to prove that M must be positive semi-definite, i.e., $z^T M z \geq 0, \forall z \in^n$

$$M = ...$$

---

b)

...

---

c)

We want to prove that N may not be unique in $M = N^T N$. For this, lets consider the orthogonal matrix Q and the new matrix that is the result by its multiplication with N, i.e., QN:

$$M = (QN)^T (QN) = (N^T Q^T)(QN) = N^T (Q^T Q) N$$

Since Q is orthogonal, $Q^T Q$ is equal to the identity matrix I. Hence:

$$M = N^T (Q^T Q) N = N^T I N = N^T N$$

So if M can be decomposed in $N^T N$, it can also be decomposed by $(QN)^T (QN)$, with Q being an orthogonal matrix.

d)

...

Question 4

The Taylor series expansion of a function $f(x, y)$ in a neighborhood around (x_0, y_0) is as follows:

$$f(x,y) \approx f(x_0,y_0) + \underbrace{f_x(x_0,y_0)(x-x_0) + f_y(x_0,y_0)(y-y_0)}_{\text{first order term}} + ...$$

Ignoring terms from second order and higher, and considering the linearized function given $f_L(x, y) = 2x + py - 8$, we have:

$$f_L(x,y) = f(x_0,y_0) + f_x(x_0,y_0)(x-x_0) + f_y(x_0,y_0)(y-y_0)$$

Taking $f(x, y) = x\sqrt{y}$, we have:

- $f_x(x, y) = \sqrt{y}$
- $f_y(x, y) = \frac{x}{2\sqrt{y}}$

$$2x + py - 8 = \underbrace{x_0\sqrt{y}_0}_{f(x_0,y_0)} + \underbrace{\sqrt{y}_0}_{f_x(x_0,y_0)}(x-x_0) + \underbrace{\frac{x_0}{2\sqrt{y}_0}}_{f_y(x_0,y_0)}(y-y_0)$$

$$2x + py - 8 = x_0\sqrt{y}_0 + x\sqrt{y}_0 - x_0\sqrt{y}_0 + \frac{x_0 y}{2\sqrt{y}_0} - \frac{x_0 y_0}{2\sqrt{y}_0}$$

$$2x + py - 8 = \sqrt{y}_0 x + \left(\frac{x_0}{2\sqrt{y}_0}\right) y - \left(\frac{x_0 y_0}{2\sqrt{y}_0}\right)$$

Now, if we compare the terms that only depend on x, that only depend on y and the independent terms (that only depend on x_0 and y_0), we have:

- (i) $2 = \sqrt{y}_0 \implies \boxed{y_0 = 4}$
- (ii) $p = \left(\frac{x_0}{2\sqrt{y}_0}\right)$
- (iii) $8 = \left(\frac{x_0 y_0}{2\sqrt{y}_0}\right)$

Substituting in (i) in (iii):

$$\frac{x_0 y_0}{2\sqrt{y}_0} = 8$$

$$\frac{4x_0}{2 \cdot 2} = 8$$

$$\boxed{x_0 = 8}$$

Substituting in (ii):

$$p = \frac{x_0}{2\sqrt{y}_0}$$

$$p = \frac{8}{2 \cdot 2}$$

$$\boxed{p = 2}$$

So the point from where the function was linearized is $(x_0, y_0) = (8, 4)$ and the coefficient $p = 2$, then $f_L(x, y) = 2x + 2y - 8$.

Question 5

Dedução:

- **Explicação:**

Na inferência dedutiva a conclusão sempre está contida nas premissas. Ao derivar uma afirmação y partindo de x, dizemos que y é uma consequência lógica de x, ou seja, a inferência de y é baseada no que foi assumido em x e será sempre verdadeira. Também podemos incluir casos particulares que são inferidos a partir de premissas universais e gerais.

- **Vantagem:**

Dos três tipos de inferência esta é a única cuja conclusão lógica é sempre verdadeira.

- **Desvantagem:**

É a única das inferências que não acrescenta conhecimento novo.

- **Exemplo:**

Se um pesquisador estiver treinando uma rede neural e por algum motivo as informações referentes ao estado atual do treinamento seja perdido (seja por falta de energia, Colab retomando a máquina virtual, ou qualquer outro) sem que haja um backup dos pesos e parâmetros, será necessário reiniciar o treinamento do zero. Logo, o pesquisador opta por salvar checkpoints do status do treinamento (pesos e outros parâmetros), para caso isso ocorra, o treinamento não seja inteiramente perdido e o pesquisador possa continuar do último estado salvo. Essa ação preventiva é tomada baseada em uma inferência dedutiva.
1. Se cai a energia e não tenho backup $\implies$ perco o treinamento
2. Quando perco o treinamento (sem backup) $\implies$ recomeço do zero
conclusão: se cai energia sem backup, perco tempo recomeçando do zero.
precaução: vou salvar checkpoints do treinamento.

Indução:

- **Explicação:**

Diferente da dedução aqui não temos mais a garantia de veracidade a cerca de inferência. Em indução temos a intuição de probabilidade, agregando experiência e conhecimento. A partir de um certo conjunto de elementos conhecidos ou observáveis, e portanto de casos particulares, chega-se em uma conclusão geral. Assim, uma característica marcante deste tipo de inferência é a capacidade de generalização.

- **Vantagem:**

Entre as vantagens deste modo de inferência podemos listar a capacidade de generalização mesmo sem haver uma certeza lógica e uso de experiências anteriores a cerca do que se está inferindo.

- **Desvantagem:**

Esta inferência não é 100% verdade como a dedutiva, portanto pode levar a generalizações errôneas. Por trabalhar com experiências passadas, tem potencial de acrescentar informação menor do que a abdução, pois boa parte dela foi usada como experiência para a própria inferência.

- **Exemplo:**

Ao rodar o loop de treinamento de uma rede neural, o pesquisador se depara com uma exceção Python por conta de um erro em operações envolvendo matrizes que apresenta um log de erro não direto e difícil de interpretar, não sendo muito útil portanto. Em experiências anteriores, na maior parte das vezes o erro era causado por descasamento nos shapes de tensores/arrays ou dimensões erradas. Assim, o pesquisador decide usar um debugger (ex: pdb) colocando o breakpoint imediatamente na linha antes do erro acontecer e checar os shapes e dimensões dos tensores envolvidos, pois desconfia que o erro envolva uma dessas duas coisas.

Abdução:

- **Explicação:**

Na inferência abdutiva trabalha-se com hipóteses para explicação do que foi observado. Trata-se de inferir x como explicação para y. A conclusão não é dado pela lógica, mas sim pela capacidade em justificar e argumentar a escolha da melhor hipótese dado um contexto. Mesmo que este tipo de inferência não tenha a certeza da inferência dedutiva, cria-se conclusões que nos fazem considerar a hipótese como plausível e/ou verdadeira.

- **Vantagem:**

Dos três tipos de inferência esta é a que mais pode acrescentar informação, caso a inferência seja condizente com a realidade e possa se sustentar com argumentos. Quando usada adequadamente, pode ser bastante útil informações a priori (priors) em estatística Bayesiana.

- **Desvantagem:**

Entretanto, é mais fraca do ponto de vista lógico, no sentido de não haver certeza alguma no que se está concluindo, havendo apenas formação de hipóteses que podem sustentar a análise.

- **Exemplo:**

O pesquisador está pela primeira vez trabalhando big data que requer uso intenso de GPU, horas de processamento, milhões de amostras, milhões de parâmetros ajustáveis, etc. Ele está treinando uma arquitetura do T5 da Google para traduzir sentenças do inglês para português. Ao iniciar o treinamento percebe que o tempo de processamento de uma época está na ordem de 10 horas. Como o pesquisador não dispõe de tempo para treinar por algumas épocas, decide que precisa diminuir o tempo de treinamento, mas tentando perder o mínimo de qualidade possível. Assim, o pesquisador decide tentar usar uma precisão de 16-bits ao invés da tradicional 32-bits. Essa foi uma escolha dentre inúmeras outras (abaixar batch size, aumentar learning rate, dimunuir tamanho das sequências de entrada/saída, usar um modelo menos parâmetros, alterar o scheduling factor, etc). Dentre todas as ações a se tomar, o pesquisador toma como primeiro palpite a tentativa de usar outro valor de precisão, com o pensamento de que este pode oferecer o melhor tradeoff entre custo computacional, mantendo qualidade na tradução de seu modelo.

Por quê treinar uma rede neural está associado a um processo de inferência indutiva?

No treinamento de uma rede neural, um dos principais conceitos está no aprendizado através de dados. O que procura-se no treinamento, é a distrubuição original e desconhecida que gerou os dados, para que se possa generalizar para além dos dados que se dispõe. Assim, se pudermos modelar uma distribuição, baseado nos dados de treinamento, podemos tirar conclusões a respeito de qualquer dado possível que esteja submetido àquela distribuição (se a generalização estiver boa).
Uma das definições de aprendizado de máquina, dada por Tom Mitchell (1997) é: "diz-se

que um programa de computador aprendeu de uma experiência E com respeito a uma certa tarefa T e com certa performance P, se sua performance em T medida por P aumenta com a experiência E". Nessa afirmação, pode-ser ver claramente os príncipios de inferência indutiva, havendo a questão da experiência em E e capacidade de generalização em T medida por P.
Por fim, temos exatamente a situação em que parte-se de casos particulares (dados de treinamento) a fim de se inferir o caso geral (distruição que gerou os dados / capacidade de generalização).

Question 6

O primeiro paper selecionado é de 2002 com 2173 citações. Esse paper foi publicado no JAMA (The Journal of the American Medical Association) que é um jornal da área médica com 48 publicações por ano pela AMA (American Medical Association).

title: Effects of Cognitive Training Interventions With Older Adults - A Randomized Controlled Trial year: 2002 cited by: 2173 publication: JAMA reference:

Ball K, Berch DB, Helmers KF, et al. Effects of Cognitive Training Interventions With Older Adults: A Randomized Controlled Trial. JAMA. 2002;288(18):2271–2281. doi:10.1001/jama.288.18.2271

O segundo paper selecionado é de 2009 e conta com 315 citações. Esse paper foi publicado no jornal acadêmico Alzheimer's & Dementia, que conta com publicações mensais da associação sem fins lucrativos Journal of the Alzheimer's Association.

title: Immediate and delayed effects of cognitive interventions in healthy elderly: A review of current literature and future directions year: 2009 cited by: 315 publication: Alzheimer's & Dementia (Volume 5, Issue 1, January 2009, Pages 50-60) reference:

Papp K V, Walsh S J, Snyder P J. Immediate and delayed effects of cognitive interventions in healthy elderly: a review of current literature and future directions. Alzheimer's and Dementia 2009; 5(1): 50-60. [PubMed]