Universidad Técnica Federico Santa María
Departamento de Informática

Introducción a las Redes Neuronales Artificiales
Cuestionario 2 - I - 2019

1. Explique en qué consiste la técnica denominada *Dropout* y qué problema intenta resolver.

2. Considere una capa convolucional que toma volúmenes de forma $32 \times 32 \times 64$ y produce volúmenes de tamaño 29×29 implementando convoluciones válidas (sin padding). Si se aplica *Dropout* a esta capa, indique qué dimensiones tienen las máscaras estocásticas utilizadas. Indique además, por qué constante deben escalarse los pesos de cada neurona si se emplea un nivel de *Dropout* igual a $p = 0.75$ (probabilidad de retener la neurona).

3. Explique en qué consiste la técnica denominada *Batch Normalization* y qué problema intenta resolver.

4. Suponga que, para entrenar una red convolucional muy profunda, usted dispone de muy pocos ejemplos etiquetados, pero de muchísimas imágenes no etiquetadas obtenidas del dominio de aplicación. ¿Indique cómo podría utilizar estas imágenes sin etiqueta para mejorar el entrenamiento de su red?

5. Explique en qué consiste la técnica denominada *data augmentation* y qué problema intenta abordar. De un ejemplos de operaciones de este tipo que podrían ser aplicadas sobre un dataset visual (imágenes) y sobre un dataset textual (e.g. opiniones).

6. Considere una pequeña red neuronal recurrente tipo Elman con input $x_t \in \mathbb{R}$, capa oculta $z_t = \alpha_1 x_t + \beta_1 z_{t-1} + b_1$ y salida $y_t = \alpha_2 z_t + b_2$. Suponga que después del entrenamiento, los pesos de la red son $\alpha_1 = \beta_1 = \alpha_2 = 1$ y $b_1 = b_2 = 0$. Haga un diagrama de la red indicando claramente los ciclos y pesos correspondientes a estas ecuaciones. Genere además una representación completamente "desenrollada" (unfolded) de la red en el tiempo para la secuencia $x_1 = 2$, $x_2 = -0.5$, $x_3 = 1$, $x_4 = 1$. ¿Qué hace esta pequeña red?

7. Derive explícitamente las ecuaciones correspondientes al back-propagation en el tiempo (BPTT) para una red neuronal diseñada para procesar una secuencia de la forma $x_1, x_2, \ldots x_T$, $x_t \in \mathbb{R}^d$ y cuya secuencia de salida $y_1, y_2, \ldots y_T$, $y_t \in \mathbb{R}^k$ viene definida mediante las siguientes ecuaciones:

$$y_t = g_o(Vh_t + c) \tag{1}$$
$$h_t = g_h(Wh_{t-1} + Ux_t + b)\,, \tag{2}$$

   con $U \in \mathbb{R}^{md}$,$W \in \mathbb{R}^{km}$, $U \in \mathbb{R}^{mk}$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^k$.

8. Suponga que se implementa una red neuronal feed-forward para procesar secuencias de largo fijo $T = 10000$, con elementos $x_t \in \mathbb{R}^{10}$. Suponga que utiliza una única capa oculta de tamaño 100 y que la salida es una secuencia del mismo largo que la secuencia de entrada, producida de manera sincronizada . ¿Cuántos parámetros tiene este modelo con respecto a una red recurrente estándar (pregunta 2) con la misma cantidad de neuronas ocultas diseñada para procesar la secuencia? Comente sobre las implicancias computacionales y estadísticas de esta diferencia.

9. Suponga que se desea entrenar una LSTM de 10 celdas que permita anticipar los niveles de una docena de contaminantes atmosféricos en la ciudad de Santiago. Se dispone de datos horarios de esos contaminantes (24 registros por día) para el período que va desde 2006 al 2016. Si representamos esta serie de datos por $(\mathbf{x}_t)_t$, $x_t \in \mathbb{R}^{12}$, indique cómo formaría las secuencias de entrenamiento y qué forma concreta tendría el arreglo de datos que usaría para entrenar el modelo. Indique además los cambios de forma que experimenta el arreglo de entrada hasta que sale de la red.

10. Explique conceptualmente porqué el problema del desvanecimiento o explosión de los gradientes se puede manifestar en redes neuronales recurrentes (RNN), aún si estas tienen una sola capa oculta. Explique además porqué este problema pone límites prácticos al largo de las secuencias que este tipo de redes pueden procesar.

11. En pocas palabras, explique cuál es la "estrategia" implementada por redes ESN (Echo state nets) para permitir el aprendizaje

12. Explique en qué consiste una "unidad permeable" (leaky unit) en el contexto de redes neuronales recurrentes y qué problema se intenta resolver mediante su introducción.

13. Dibuje una típica célula de memoria LSTM que represente claramente el rol de las puertas de entrada, salida y olvido, así como la entrada y salida final del bloque. Escriba además, las ecuaciones que permiten definir formalmente el comportamiento del bloque en el tiempo t, denotando por x_t al patrón de entrada, h_t al patrón de salida y por i_t, o_t, f_t a las activaciones de las compuertas de entrada, salida y olvido respectivamente. Finalmente, explique las ventajas de esta arquitectura con respecto a una red neuronal recurrente clásica.

14. Explique cuál es la diferencia entre una red GRU respecto de una LSTM. ¿Qué objetivo se persigue con esta modificación?

15. Considere una red LSTM con compuertas estándares de entrada, salida y olvido diseñada para procesar una secuencia $x_1, x_2, \ldots x_T$. Construya un diagrama que indique como debiesen organizarse las compuertas de una celda en modo de transmitir intacto un elemento x_{t_1} de la secuencia hasta un tiempo $t_2 \gg t_1$.

16. Considere una red neuronal recurrente de 1 capa oculta, con recurrencias desde la salida (tipo Jordan), entrenada para producir una secuencia a partir de otra secuencia (many-to-many). Una técnica usada con frecuencia en este caso, consiste en entrenar la red usando *teacher forcing*, es decir, usando la salida correcta en cada tiempo como input para el tiempo sucesivo, en vez de la predicción de la red.

    (a) Explique porqué este criterio es óptimo si se adopta el criterio de estimación de máxima verosimilitud.

    (b) Explique porqué podría ser inconveniente entrenar a la red usando las salidas correctas en vez de sus propias predicciones y proponga un criterio para atenuar este efecto.

17. Suponga que desea entrenar una red LSTM usando *Dropout* sobre los pesos recurrentes. Para una determinada secuencia, ¿Debe ser la máscara correspondiente compartida/fija en el tiempo? o ¿puede aplicarse una realización diferente en cada instante de tiempo?

18. ¿Qué es un modelo neuronal de lenguaje? ¿Porqué se utilizan redes recurrentes para implementar dichos modelos? Cierre su respuesta mencionando una aplicación interesante de los mismos.

19. Explique qué distingue un modelo de aprendizaje supervisado de uno no supervisado, indicando al menos 3 aplicaciones de este último.

20. Explique qué distingue un modelo de aprendizaje generativo de uno discriminativo, indicando al menos 1 aplicación de este último.

21. Explique brevemente qué es un auto-encoder. ¿Es cierto o es falso que para propósitos prácticos sólo es relevante el encoder y que el decoder se mantiene activo sólo durante la fase entrenamiento? Justifique.

22. ¿Es cierto o es falso que el sub-espacio extraído por PCA es equivalente (hasta transformaciones lineales) al espacio latente aprendido por un auto-encoder lineal? ¿Depende esto de cómo se mida el error de reconstrucción?

23. Sea $p(x, z)$ una distribución de probabilidad válidamente definida sobre el par de variables (x, z). ¿Es cierto o falso que para cualquier distribución de probabilidad $Q(z)$ vale la siguiente desigualdad?

$$\log\left(\sum_z p(x,z)\right) \leq \sum_z Q(z)\log\left(\frac{p(x|z)p(z)}{Q(z)}\right) \tag{3}$$

Si su respuesta es negativa, corrija la desigualdad. En cualquier caso, justifique su respuesta, indicando además en qué escenario resulta tan utilizado el resultado obtenido.

24. Explique en qué consiste el algoritmo de Gibbs y cómo viene utilizado durante el entrenamiento de RBMs. ¿Está asegurada su convergencia? ¿Cuál es su ventaja sobre un método más genérico como el algoritmo de Metropolis?

25. ¿Cuál es la simplificación fundamental que hace el algoritmo CD-k para entrenar RBMs? (la abreviación CD corresponde al término *contrastive divergence*).

26. Considere el modelo $p(x, z) = \exp(-E(x, z))/Z$, con $E(x, z) = -(z^T W x + b^T z + c^T x)$, $x \in \mathbb{R}^d$, $z \in \{0, 1\}^d$, definido por una RBM. Demuestre que

$$p(z = 1|x) = \frac{\exp(Wx + b)}{1 + \exp(Wx + b)} = \sigma(Wx + b)\,, \tag{4}$$

27. Considere una RBM como aquella definida en la pregunta 8 y la función de log-verosimilitud asociada a un punto $x \in \mathbb{R}^d$, $\ell(W) = \ln p(x|W)$, donde W representa los parámetros del modelo. Demuestre que

$$\frac{\partial \ell(W)}{\partial W_{ji}} = \mathbb{E}_{h|x} h_j x_i - \mathbb{E}_{x,h} h_j x_i \tag{5}$$

¿Porqué el segundo término es mucho más difícil de calcular que el segundo? ¿Como se aborda este problema en el algoritmo CD-k?

28. Considere un auto-encoder donde el mecanismo generador se modela como $z \sim p_\theta(z)$ y luego $x \sim p_\theta(x|z)$. ¿Porqué resulta difícil calcular p_θ(z—x)? ¿Cómo se aborda este problema en el diseño de auto-encoders variacionales (VAEs)?

29. Explique en qué consiste una capa convolucional traspuesta (transposed convolution) y cómo se implementa en la práctica. Explique además cómo es utilizada en el diseño de modelos generativos de datos visuales.

30. Considere un auto-encoder variacional donde el encoder se modela mediante la distribución $q_\phi(z|x)$, el decoder mediante la distribución $p_\theta(x|z)$ y el a-priori sobre el espacio latente mediante la distribución $p_\theta(z)$. ¿Porqué resulta conveniente elegir $q_\phi(z|x)$ y $p_\theta(z)$ gausianas?

31. Explique la diferencia entre la divergencia Kullback-Leibler $KL(p||q)$ y la divergencia de Jensen-Shannon $JS(p||q)$. ¿Cómo trata cada métrica los puntos x a los que $p(x)$ asigna probabilidad 0? Reflexione sobre las consecuencias de esta observación.

32. Considere un auto-encoder variacional como el de la pregunta 12. Como hemos visto, entrenar un auto-encoder variacional resulta equivalente a maximizar la siguiente función objetivo,

$$J(\phi, \theta) = \sum_\ell p_\theta(x^{(\ell)}) - KL(q_\phi(z|x^{(\ell)})||p_\theta(z|x^{(\ell)})) \tag{6}$$

Mencione una consecuencia positiva y una consecuencia negativa de cambiar la divergencia KL por una divergencia de Jensen-Shannon.

33. Considere un auto-encoder variacional como el de la pregunta 12. Como hemos visto, entrenar un auto-encoder variacional resulta equivalente a maximizar

$$J(\phi, \theta) = \mathbb{E}_{x \sim p(x)} p_\theta(x^{(\ell)}) - KL(q_\theta(z|x^{(\ell)})||p_\theta(z|x^{(\ell)}))\,, \tag{7}$$

donde $p(x)$ es la distribución real de los datos. Demuestre que esta función objetivo es equivalente a maximizar

$$J'(\phi, \theta) = \mathbb{E}_{z \sim q_\phi(z|x)} \log p_\theta(x|z) - KL(q_\theta(z|x^{(\ell)})||p_\theta(z)) \tag{8}$$

Interprete cada miembro de esta última función objetivo.

34. Considere un auto-encoder variacional como el de la pregunta 12 y la función objetivo $J'(\phi, \theta)$ definida en la pregunta 15. Explique cómo se calculan las derivadas de $J'(\phi, \theta)$ en función de ϕ y θ en modo de obtener estimadores estables. Explique la relevancia de este "truco" (denominado en ocaciones el "reparametrization trick") en el éxito experimental de los VAE.

35. Explique en qué consiste un "juego a suma cero" y un "equilibrio de Nash" en la teoría clásica de juegos.

36. ¿Cuáles son los componentes fundamentales en el modelo GAN (generative adversarial training)? ¿Cuál es el rol de cada componente? ¿Es correcto afirmar que en este modelo, el resultado del aprendizaje representa un "equilibrio de Nash" para cierto juego continuo a suma cero? Si esto último es correcto, cuál es la función de valor correspondiente.

37. Considere un juego continuo a suma cero, con dos jugadores, espacios de acciones $X = Y = \mathbb{R}$ y función de valor $U = xy$, $x \in X$, $y \in Y$. Suponga que el primer jugador (que desea maximizar U) actualiza iterativamente su estrategia (x) de la forma $x \leftarrow x + \frac{\partial U}{\partial x}$ y que el segundo jugador (que desea minimizar U) actualiza iterativamente su estrategia (y) de la forma $y \leftarrow y - \frac{\partial U}{\partial x}$. Es claro que un punto silla (equilibrio) de U es el punto $(x^*, y^*) = (0, 0)$. Muestre que si modelamos las actualizaciones ejecutadas por los jugadores de manera continua en el tiempo t jamás se converge a (x^*, y^*). ¿Qué sugiere este resultado?

Hint: Si modelamos las actualizaciones ejecutadas por los jugadores de manera continua en el tiempo t, obtenemos las siguientes ecuaciones diferenciales:

$$\frac{\partial x}{\partial t} = \frac{\partial V}{\partial x}, \quad \frac{\partial y}{\partial t} = -\frac{\partial V}{\partial x}\ . \tag{9}$$

38. Considere una GAN con discrminador $D(x)$ y generador $p_g(x) = G(z)$, $z \sim p_z(z)$. Demuestre que si asumimos que el discriminador puede siempre implemnentar su solución óptima, entrenar el generador es equivalente a minimizar

$$C(G) = 2 \cdot \mathrm{JS}(p(x)||p_g(x)) - \log(4)\,.$$

donde $\mathrm{JS}(p(x)||q(x))$ es la divergencia de Jensen-Shannon entre $p(x)$ y $q(x)$.

39. Considere una GAN con discrminador $D(x)$ y generador $p_g(x) = G(z)$, $z \sim p_z(z)$. Demuestre que para cualquier discriminador fijo $D^*(x)$, la función objetivo del lado del generador

$$\min_G C_{D^*}(G)\,, \tag{10}$$

con

$$C_{D^*}(G) = \mathbb{E}_{x \sim p(x)} \left[\log D^*(x)\right] + \mathbb{E}_{z \sim p_z(z)} \left[\log(1 - D^*(G(z)))\right] \tag{11}$$

acepta como solución óptima cualquier función $G(\cdot)$ que mapee $z \sim p_z$ a alguna de las modas de $D^*(x)$. Reflexione sobre las implicancias de este resultado.

40. Dibuje cómo arquitectuaría una red neuronal para contar el número de personas en una imagen.

RNA+CVV LaTeX