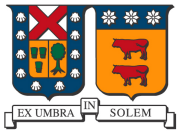


Universidad Técnica Federico Santa María
Departamento de Informática

Trabajo - Redes Neuronales Artificiales Avanzadas

Prof. Ricardo Ñanculef

Autor:
Francisco Mena Toro
201373504-5

Problema

El reconocer ciertas características de los datos a través de la probabilidad de que pertenezca a cada una de las categorías presentes en algún dominio. Asumiendo que los datos modelan correctamente la probabilidad de que pertenezca a cada una de las clases, es decir, la técnica *majority voting* resulta ser la clasificación real (*ground truth*) del dato.
La tarea consiste en imitar la distribución de probabilidad del dato i para todas las categorías j, $\hat{p}_{ij}$, es decir, no solo en la categoría más probable, sino que toda la distribución, aproximando la incerteza del dato a cada categoría, incluso las menos probables (evitando asignar probabilidad 0 a éstas si es que no lo son, $\hat{p}_{ij} \neq 0$. Todo ésto de la mano con lograr un rápido entrenamiento.

Como solución al problema se busca probar distintas funciones objetivos, en particular se trabajará en el dominio de las *bregman divergences* [Vemuri et al., 2011] y verificar cuál se comporta mejor para el problema.

Hipótesis: Una función evaluadora (función objetivo) basada en distribuciones de probabilidad logra imitar mejor el comportamiento que una función estándar de variables continuas.

Funciones Objetivos

Considerando la variable p_i como la probabilidad real observada de un dato i sobre las K categorías indexadas a través de j de manera que p_{ij} considera la probabilidad del dato sobre la categoría j, además que $\hat{p}_{ij}$ es la predicción del modelo.
A continuación detallamos las funciones objetivos utilizadas para comparar, las cuales son evaluadas por cada ejemplo en un *batch* y fusionadas a través de un promedio aritmético, como un modelo estándar de aprendizaje de máquina.

Keras y otros

En primer lugar se definen algunas comunmente utilizadas en Keras y otras a evaluar.

- *Mean Squared Error (MSE)*:

$$MSE(p_i, \hat{p}_i) = \frac{1}{K} \sum_{j}^{K} (p_{ij} - \hat{p}_{ij})^2 \tag{1}$$

- *Root Mean Squared Error (RMSE)*:

$$RMSE(p_i, \hat{p}_i) = \sqrt{\frac{1}{K}\sum_{j}^{K}(p_{ij} - \hat{p}_{ij})^2} \tag{2}$$

- *KL Reverse*:

$$KL(\hat{p}_i||p_i) = \sum_{j}^{K} \hat{p}_{ij} \log \frac{\hat{p}_{ij}}{p_{ij}} \tag{3}$$

- *Cross-entropy*

$$H(p_i, \hat{p}_i) = \sum_{j}^{K} -p_{ij} \log \hat{p}_{ij} \tag{4}$$

- *Jensen Shanon divergence* (también *symetric KL*):

$$symmetric\ KL(p_i, \hat{p}_i) = \sum_{j}^{K} KL(p_{ij}||\frac{p_{ij} + \hat{p}_{ij}}{2}) + KL(\hat{p}_{ij}||\frac{p_{ij} + \hat{p}_{ij}}{2}) \tag{5}$$

Bregman Divergences

En segundo lugar se definen las funciones de divergencia (inverso a similaridad) de Bregman [Chen et al., 2008], [Banerjee et al., 2005] para medir la diferencia entre dos distribuciones de probabilidad. Estas funciones son un grupo de familia que comparte algunas propiedades ya que son derivadas de una estructura/*framework* general.
Sea Φ una función estrictamente convexa y diferenciable, se tiene la divergencia de Bregman d_Φ definida como:

$$d_\Phi(x, y) = \Phi(x) - \Phi(y) - \langle x - y, \nabla(y) \rangle \tag{6}$$

Con $\langle \cdot, \cdot \rangle$ el *inner product*. Como se puede ver, el orden en que se entrega los argumentos importa, por lo que ésto no es una métrica ya que no cumple simetría ni desigualdad triangular. Sin embargo, tiene otras propiedades que benefician la optimización:

- Convexa: $d_\Phi(x, y)$ es convexa en su primer argumento x.
- No negativa: $d_\Phi(x, y) \geq 0$ para todo x, y.
- Dualidad: Si Φ tiene una conjugada convexa puede ser utilizada.
- La media como mínimo en aleatoriedad: Dado un conjunto de vectores aleatorio, el mínimo en $d_\Phi(x, y)$ para y, dado cualquier función Φ y algún x, es la media de esos vectores.

Entonces, la divergencia $d_\Phi(p_i, \hat{p}_i)$ con las distintas funciones utilizadas queda como:

- *Square Euclidean Distance* (también *sum of square error/sse)*, para un $\Phi(p_i) = ||p_i||^2$, (cuadrado de la norma) se tiene la divergencia de bregman:

$$SSE(p_i, \hat{p}_i) = \sum_{j}^{K}(p_{ij} - \hat{p}_{ij})^2 \tag{7}$$

- *KL Forward*, para un $\Phi(p_i) = \sum_j^K p_{ij} \log p_{ij}$ (entropía negativa), se tiene la divergencia de bregman:

$$KL(p_i \,||\, \hat{p}_i) = \sum_{j}^{K} p_{ij} \log \frac{p_{ij}}{\hat{p}_{ij}} \tag{8}$$

- *Generalized I divergence*, similar a *KL Forward* pero generalizada a un dominio de los reales positivos[1].

$$GeneralizedI(p_i \mid\mid \hat{p}_i) = \sum_j^K p_{ij} \log \frac{p_{ij}}{\hat{p}_{ij}} - (p_{ij} - \hat{p}_{ij}) \tag{9}$$

- *Itakura Saito distance*, para un $\Phi(p_i) = -\log p_i$, se tiene la divergencia de bregman:

$$ItakuraS(p_i, \hat{p}_i) = \sum_j^K \frac{p_{ij}}{\hat{p}_{ij}} - \log \frac{p_{ij}}{\hat{p}_{ij}} - 1 \tag{10}$$

Métricas de desempeño

Para compara el efecto de las distintas funciones objetivos se utilizarán las siguientes medidas de desempeño.

- Delta de convergencia.

$$\Delta(t) = \frac{\mid loss^{(t)} - loss^{(t+1)} \mid}{loss^{(t)}} \tag{11}$$

  con t el instante de tiempo durante el entrenamiento, análogo a los *epochs*.

- Top k accuracy: el modelo entrega las k categorías más probables y se verifica con la categoría mayoritaria real. Top-1 es *majority vote* en la predicción.

- *Macro* F1 score:

$$F_1^M = \frac{1}{K} \sum_j^K F_1(j) \tag{12}$$

  Calculado sobre cada clase j, $F_1(j)$, y tomando un promedio sobre éstas.

- *Normalized discounted cumulative gain* (NDCG), una métrica de *learning to rank* para medir el orden de las probabilidades predichas.

- *Accuracy on Ranking Decrease*:

$$accuracy_{ranking} = \frac{1}{N} \sum_i^N \frac{1}{K} \sum_k^K \frac{I(y_i^{(k)} = \hat{y}_i^{(k)})}{k} \tag{13}$$

  Con $y_i^{(k)}$ la categoría real del dato i en la posición k.

Experimentación

Las pruebas fueron realizadas por redes neuronales estándar, entrenando el modelo con cada función objetivo 4 veces para neutralizar la aleatoriedad del optimizador utilizado (*Adam*) y la inicialización de los pesos (*Glorot Uniform*), se utilizó tamaño de *batch* de 128 y límite de *epochs* de 20.

[1] *KL Forward* está para un dominio de valores discretos

Datos

1. GalaxyZoo[2], es un proyecto iniciado por astrónomos en la Universidad de Oxford en 2007 en donde preguntaron a personas en la vía pública (ahora en [3]) que clasificaran su dataset de un millón de galaxias (gracias a SDSS[4]). El dataset trabajado es un pequeño extracto de ésto con la probabilidad de la clasificación sobre distintas morfologías (formas) de la galaxia a través de miles de voluntarios (anotadores).

   Se trabaja con 10000 imágenes RGB redimensionadas a 100x100 pixeles, proporcionado a través de las 60 mil imágenes de 424x424 en Kaggle. Las etiquetas, también son un extracto, corresponden a la probabilidad de las respuestas a 7 preguntas de la morfología presente en la imagen, las cuales consideramos como categorías:

   a) How round is the smooth of the galaxy? Completetly round
   b) How round is the smooth of the galaxy? between
   c) How round is the smooth of the galaxy? Cigar shaped
   d) What type of disk is the galaxy? A view edge-on disk
   e) What type of disk is the galaxy? Spiral tight
   f) What type of disk is the galaxy? Spiral medium
   g) What type of disk is the galaxy? Spiral loose
   h) What type of disk is the galaxy? Normal disk
   i) Is it a Galaxy? Is a Star or artifact

   Donde lo más probable en el dataset resulta ser *Normal disk* y *Smooth between*.

Arquitectura utilizada

El modelo de red convolucional utilizada es un modelo estándar de 3 bloques convolucionales, del tipo $C \times P$, con filtros de 3×3 y números de filtros 32, 64 y 128 respectivamente, con dos capas densas al final de 512 neuronas y funciones de activación *relu* para todas las capas. Una arquitectura similar es presentada por el ganador de la competencia en la Figura 1.

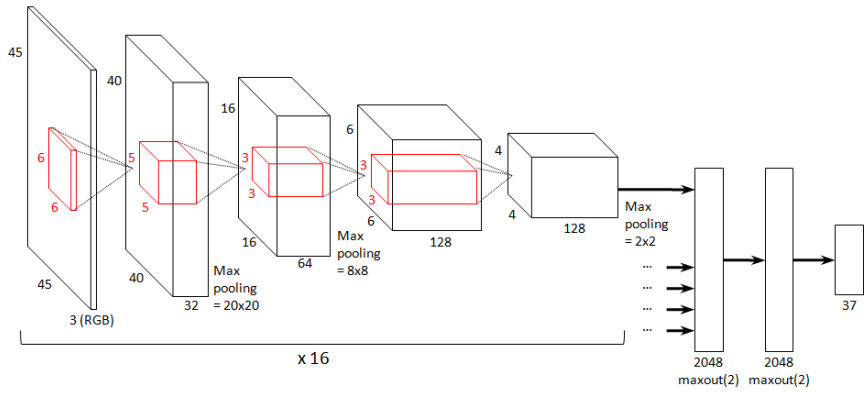


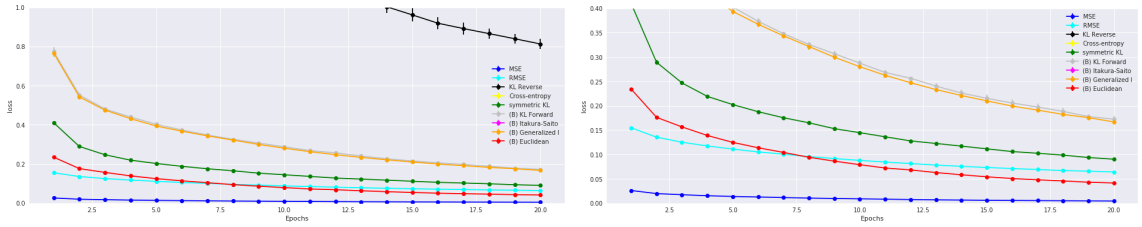
Figura 1: Red convolucional de la ganadora del challenge GalaxyZoo en Kaggle.

[2] www.galaxyzoo.org
[3] www.galaxyzoo.org/#/classify
[4] www.sdss.org

2. *Stock tweets emotion*[5] es un dataset de texto en que se tienen múltiples anotaciones de personas a *tweets* en portugués sobre la emoción de éste. Trabajando con una red recurrente estándar GRU de 2 capas con una densa al final que entrega la etiqueta de la probabilidad de las emociones: *joy*,*sadness*, *trust*, *disgust*, *surprise*, *anticipation*, *anger*, *fear* y *neutral*, siendo esta última las más probable en todo el dataset.

Resultados y Análisis

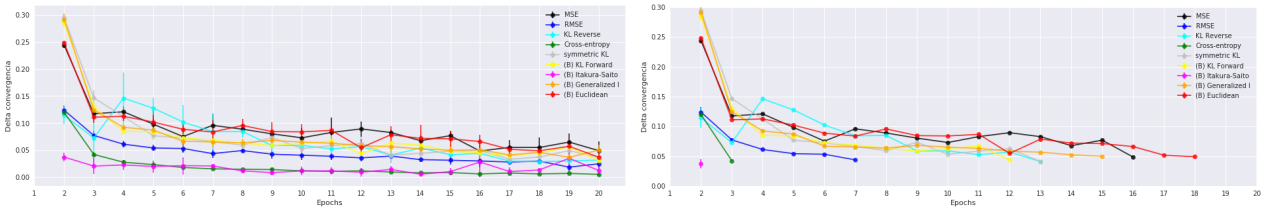
Como resultados del efecto de las distintas funciones objetivos sobre el dataset de GalaxyZoo se muestran en los gráficos de convergencia de los algoritmos (Ecuación 11) debido a las distintas escala las funciones objetivos, graficarlos unos con otros no tiene mucho sentido. Sin embargo, se puede ver el comportamiento general, en la Figura 2, algunos comentarios de ésto es que la escala de la métrica *Itakura Saito* es mucho mayor al resto (de magnitud cerca de 50), mientras que la escala de las métricas numéricas de regresión como *mse*, *rmse* y *euclidean* resultan estar en una escala muy baja, menor a 0,3. Otra observación es que el comportamiento de *Kl forward* es practicamente igual al de *Generalized I*, mientras que *cross entropy* y *symmetric KL* poseen una curvatura de evolución de función objetivo similar a la de éstas dos anteriores nombradas. Por otro lado la curvatura de la evolución de la función objetivo de *MSE* resulta similar a la de *RMSE*. Ésto se debe a la forma de las funciones objetivos, ya que son similares, exceptuando por algunas constante multiplicativa.



(a) Límite loss = 1 (b) Límite loss = 0,4

Figura 2: Evolución de funciones de pérdida utilizadas a través del número de epochs.

En la Figura 3 se puede ver el análisis de convergencia de las funciones de pérdida, en la Figura 3.a que las funciones objetivos utilizadas convergen de distinta manera, en 3.b se puede ver en mayor detalle que la función de pérdida *Itakura Saito* es la primera en dejar de variar (converge rápido). En segundo lugar la sigue la función objetivo *cross entropy*, convergiendo en el tercer *epoch*, la sigue *RMSE* convergiendo en el séptimo *epoch*. Luego de ésto converge *KL forward* en el *epoch* 12, seguido de KL reverse y symmetric KL en el *epoch* 13. Los últimos en converger al límite 0.05 de variación, pasado los 15 *epoch*, son *Generalized I*, *MSE* y *Euclidean*.



(a) Durante todo el entrenamiento (b) Detenido si delta es menor a 0,05

Figura 3: Evolución del delta de convergencia a través del número de epochs.

En la Tabla 1 se muestra el resultado de la métrica *macro F1* en ambos conjuntos de datos para la clasificación sobre la categoría mayoritaria (*majority voting*), donde se puede ver que los peores resultan ser con

[5]https://www.kaggle.com/fernandojvdasilva/stock-tweets-ptbr-emotions

B	Funciones objetivos	*training*	*test*
	MSE	**83,910**	47,747
	RMSE	75,773	40,804
	KL Reverse	38,107	29,519
	Cross-entropy	80,075	45,352
	symmetric KL	75,486	41,543
✓	*KL Forward*	79,144	46,005
✓	*Itakura-Saito*	16,584	17,067
✓	*Generalized I*	78,944	**48,934**
✓	*Squared Euclidean*	80,333	39,156

Tabla 1: Resultados en base a métrica *macro F1* en ambos conjuntos, para las distintas funciones objetivos, en porcentaje. *B* hace referencia a las *bregmann divergences*.

B	Funciones objetivos	*training*	*test*
	MSE	96,847	**94,708**
	RMSE	96,823	94,559
	KL Reverse	93,914	93,080
	Cross-entropy	**97,476**	**94,711**
	symmetric KL	**97,276**	**94,955**
✓	*KL Forward*	**97,403**	94,580
✓	*Itakura-Saito*	91,867	91,805
✓	*Generalized I*	**97,364**	**94,779**
✓	*Squared Euclidean*	96,824	94,462

Tabla 2: Resultados en base a métrica NDCG (para todo las clases $topk = K$) en ambos conjuntos, para las distintas funciones objetivos. Valores en porcentaje y negrita los 4 mejores en cada set.

Itakura Saito seguido de *KL Reverse*. La función objetivo que logra generalizar mejor en base a esta métrica resulta ser *Generalized I*, luego está *MSE*, *KL forward* y *cross entropy*, en ese orden.
Las funciones objetivos que mejor se comportan en el conjunto de entrenamiento, son *MSE* seguido de *cross entropy* y *euclidean*, pudiendo ver que se repite *MSE* y *cross entropy* en los mejors resultados.

En la Tabla 2 se muestra el resultado de la métrica de ranking *NDCG*, para evaluar el orden en que entregan las categorías de cada dato en base a sus probabilidades. Similar al caso anterior, los peores valores se encuentran en la función objetivo *Itakura Saito* y *KL reverse*, mientras que los mejores (marcados en negritas) resultan similares en entrenamiento y pruebas, siendo *Generalized I*, *cross entropy* y *symmetric KL*, además de que *MSE* se comporta bien en pruebas. Para este caso, en que se evalúa en que orden se entregan las categorías en base a su probabilidad, las que mejor resultan son justamente las funciones objetivos que están basadas en etiquetas probabilistas, por lo que si ese fuera el objetivo, este tipo de funciones resultan óptimas.

Como métrica alternativa para evaluar el cómo el modelo entrega el orden de las probabilidades de los datos, se midió *Accuracy on Ranking Decrease* (Ecuación 13), mostrando su resultado en la Tabla 3. La peor función objetivo vuelve a resultar con *Itakura Saito* y *KL reverse*. La función objetivo que mayor *score* logra en entrenamiento es *KL forward*, mientras que *symmetric KL* es la que generaliza mejor. Otras funciones que no se quedan atrás en esta evaluación, nuevamente resultan las basadas en probabilidades: *cross entropy* y *Generalized I*.

B	Funciones objetivos	*training*	*test*
	MSE	28,835	16,851
	RMSE	28,252	17,214
	KL Reverse	20,175	15,862
	Cross-entropy	30,652	17,267
	symmetric KL	30,625	**18,657**
✓	*KL Forward*	**31,007**	16,788
✓	*Itakura-Saito*	11,262	11,256
✓	*Generalized I*	30,659	17,276
✓	*Squared Euclidean*	28,350	16,700

Tabla 3: Resultados en base a métrica *Accuracy on ranking decrease* en ambos conjuntos, para las distintas funciones objetivos.

Se probó la arquitectura de la Figura 1 sobre este dataset y los resultados se mantuvieron. Lo que se destaca es que *symmetric KL* se destaca sobre entrenamiento y pruebas en la métrica *macro F1*.

Sobre las pruebas en el dataset de texto *Stock tweets emotions*, se encuentran que reportan resultados similares, exceptuando a *KL reverse* que se destaca por sobre todas las otras funciones objetivos, posiblemente a que el dataset es muy desbalanceado y esta función objetivo actúa como regularizador gracias a maximizar la entropía de la predicción además de minimizar la distancia entre predicción y real.

Resumiendo lo experimentado se tiene que las funciones de la familia *Bregman divergences* finalmente no se comportaron todas de buena manera como grupo, al compartir las mismas propiedades. Posiblemente como no son métricas, le falte la propiedad de simetría y desigualdad triangular para tener un mejor comportamiento. La función objetivo más utilizada para clasificación con probabilidades, *cross entropy*, resultó resaltar de buena manera. Mientras que *Generalized I*, a pesar de no ser muy estudiada o elegida, presenta un muy buen desempeño, además de generalizar mejor que el resto.
KL reverse en el dataset de Galaxyzoo no tuve el efecto esperado de poder actuar como regularizador, sin embargo, en el dataset de texto si lo logra.
A pesar de que *cross entropy* es igual a *KL forward* en optimización, exceptuando por la entropía de la probabilidad real p, $H(p)$, que no depende de la optimización en la predicción q:

$$H(p, q) = H(p) + KL(p||q)$$

$H(p)$ resulta cero en la primera derivada parcial. Sin embargo, estas dos funciones objetivos llegan a valores distintos en la optimización, a pesar de tener el mismo mínimo global, a través de la optimización estocástica *Kl forward* se queda por detrás de *cross entropy* en la experimentación, posiblemente a que la derivada funcional de Keras no sea tenga tan buena precisión. Caso similar se ve entre *MSE* y *Squared Euclidean distance*, en donde tan solo difieren en una constante, $\frac{1}{K}$, también llegan a valores distintos en cada caso. Posiblemente a que, a pesar de tener el mismo mínimo global, la curvatura de la primera derivada es distinta, amplificándola o reduciéndola a través de esa constante que se multiplica. Otra posibilidad es que sea solamente a través de la optimización estocástica realizada.

Conclusiones

En este reporte se estudiaron distintas funciones objetivos para optimizar en un problema de estimar las probabilidades de los datos, midiendo distintas métricas para evaluar la calidad de éstas.

Algunos resultados reflejan correlación a la hipótesis, como los buenos desempeño de *cross entropy*, *Generalized I*, *symmetric KL* y *KL* en base a las métricas de desempeño para el *ranking* de las probabilidades, por lo que si ese fuera el objetivo estas funciones objetivos debieran ser las ideales, ya que logran imitar de buena manera el orden en que se deben entregar las categorías de los datos, además de su probabilidad.

El resultado inesperado encontrado de que funciones objetivos analíticamente iguales en la optimización llegan a tener resultados muy distintos al final del entrenamiento en varias métricas, pueda deberse a distintos factores de programación y optimización aproximada, o bien, que los factores multiplicativos en la optimización tengan un aporte que no puede ser ignorado al eliminarlos.

Referencias

[Banerjee et al., 2005] Banerjee, A., Merugu, S., Dhillon, I. S., and Ghosh, J. (2005). Clustering with bregman divergences. *Journal of machine learning research*, 6(Oct):1705–1749.

[Chen et al., 2008] Chen, P., Chen, Y., Rao, M., et al. (2008). Metrics defined by bregman divergences: Part 2. *Communications in Mathematical Sciences*, 6(4):927–948.

[Vemuri et al., 2011] Vemuri, B. C., Liu, M., Amari, S.-I., and Nielsen, F. (2011). Total bregman divergence and its applications to dti analysis. *IEEE Transactions on medical imaging*, 30(2):475–483.