DECSAI
Departamento de Ciencias de la Computación e I.A.
Universidad de Granada

Deep Stacking Networks [DSNs]

Fernando Berzal, berzal@acm.org

Deep Stacking Networks (DSNs)

- Las técnicas convencionales para entrenar DNNs, en su etapa final de ajuste [fine tunning] requieren la utilización del gradiente descendente estocástico, que resulta difícil de paralelizar.

- La arquitectura de las DSNs (originalmente llamadas "Deep Convex Networks" o DCNs) se diseñó con el problema de la escalabilidad del aprendizaje en mente.

Li Deng & Dong Yu: "Deep convex network: A scalable architecture for speech pattern classification". Interspeech'2011.

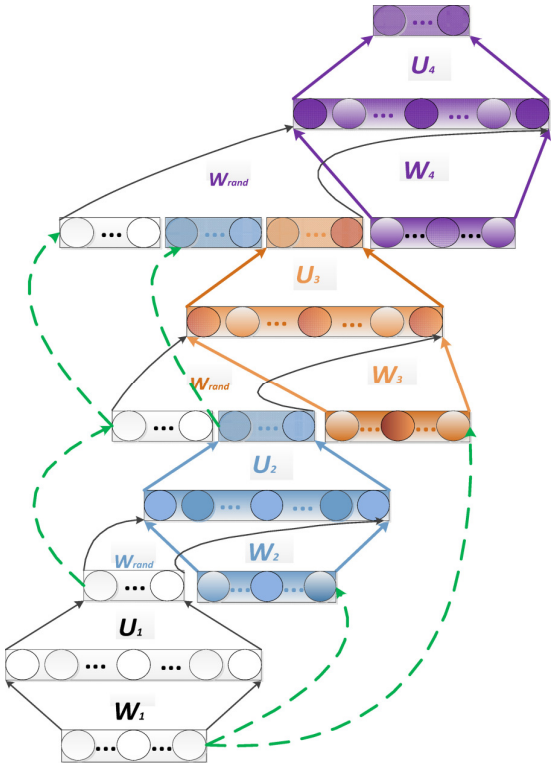
Deep Stacking Networks (DSNs)

- Las DSNs apilan redes multicapa como módulos básicos en los que las unidades de salida son lineales y las unidades ocultas son sigmoidales (no lineales).
- La linealidad de las unidades de salida permite una estimación eficiente y paralelizable de los pesos de la capa de salida dada la actividad de la capa oculta (al tratarse de un problema de optimización convexa).
- Las restricciones impuestas por la arquitectura hacen mucho más fácil el aprendizaje de los demás parámetros de la red (los pesos de la capa oculta).

Deep Stacking Networks (DSNs)

La matriz de pesos W conecta la capa lineal de entrada con la capa oculta no lineal.

La matriz de pesos U conecta la capa oculta (no lineal) con la capa lineal de salida.



Deep Stacking Networks (DSNs)

Algoritmo de aprendizaje

Dados los vectores de salida para el conjunto de entrenamiento T, los pesos U y W se ajustan para minimizar el error:

$$E = \frac{1}{2}\sum_i \|\boldsymbol{y}_i - \boldsymbol{t}_i\|^2 = \frac{1}{2}\mathrm{Tr}[(\boldsymbol{Y}-\boldsymbol{T})(\boldsymbol{Y}-\boldsymbol{T})^{\mathrm{T}}]$$

donde Y es la salida de la red:

$$\boldsymbol{y}_i = \boldsymbol{U}^{\mathrm{T}}\boldsymbol{h}_i = \boldsymbol{U}^{\mathrm{T}}\sigma(\boldsymbol{W}^{\mathrm{T}}\boldsymbol{x}_i) = G_i(\boldsymbol{UW})$$

Deep Stacking Networks (DSNs)

Algoritmo de aprendizaje

U puede aprenderse una vez que se tiene la matriz de actividad H para la capa oculta (o lo que es equivalente, cuando se conoce W).

Fijando la derivada del error con respecto a U a cero, obtenemos

$$\boldsymbol{U} = (\boldsymbol{HH}^{\mathrm{T}})^{-1}\boldsymbol{HT}^{\mathrm{T}} = \mathrm{F}(\boldsymbol{W}), \quad \text{where } \boldsymbol{h}_i = \sigma(\boldsymbol{W}^{\mathrm{T}}\boldsymbol{x}_i).$$

Esto nos proporciona una restricción explícita entre U y W que podemos aprovechar en la minimización del error.

Deep Stacking Networks (DSNs)

Algoritmo de aprendizaje

Dada la restricción U=F(W),
aplicamos el método de los multiplicadores de Lagrange:

$$E = \frac{1}{2}\sum_i \|G_i(\boldsymbol{U}, \boldsymbol{W}) - \boldsymbol{t}_i\|^2 + \lambda\|\mathbf{U} - \mathrm{F}(\boldsymbol{W})\|$$

Ahora, podemos utilizar "full-batch learning":

$$\frac{\partial E}{\partial \boldsymbol{W}} = 2\boldsymbol{X}[\boldsymbol{H}^{\mathrm{T}} \circ (\mathbf{1} - \boldsymbol{H})^{\mathrm{T}} \circ [\boldsymbol{H}^{\dagger}(\boldsymbol{H}\boldsymbol{T}^{\mathrm{T}})(\boldsymbol{T}\boldsymbol{H}^{\dagger}) - \boldsymbol{T}^{\mathrm{T}}(\boldsymbol{T}\boldsymbol{H}^{\dagger})]]$$

$$\boldsymbol{H}^{\dagger} = \boldsymbol{H}^{\mathrm{T}}(\boldsymbol{H}\boldsymbol{H}^{\mathrm{T}})^{-1}$$

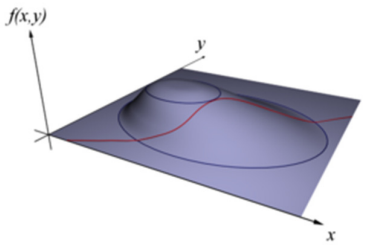
Deep Stacking Networks (DSNs)

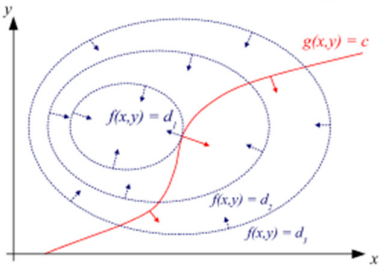
Método de Lagrange

Método de optimización de funciones
de múltiples variables sujetas a restricciones

Función objetivo $f(\theta)$

Restricción $g(\theta) = 0$





Los gradientes $\nabla f(\theta)$ y $\nabla g(\theta)$
son paralelos, por lo que

$$\nabla \mathbf{f}(\theta) = \lambda \nabla \mathbf{g}(\theta)$$

Deep Stacking Networks (DSNs)

Método de Lagrange

Podemos incorporar la restricción $g(\theta)=0$ a nuestra función objetivo, añadiendo una nueva variable λ:

$$\mathbf{F(\theta,\lambda) = f(\theta) - \lambda g(\theta)}$$

F se denomina Lagrangiano.
Optimizamos, como siempre, igualando a cero su gradiente:

$$\nabla \mathbf{F(\theta,\lambda) = 0}$$

Deep Stacking Networks (DSNs)

Método de Lagrange

EJEMPLO

Cómo minimizar la superficie de una caja de volumen V.

Función objetivo: $f(x,y,z) = 2(xy+xz+yz)$
Restricción: $g(x,y,z) = xyz - V = 0$

Gradiente $\nabla f(x,y,z) = < 2(y+z), 2(x+z), 2(x+y) >$
$\nabla g(x,y,z) = < yz, xz, xy >$

Al hacer $\nabla f(x,y,z) = \lambda \nabla g(x,y,z)$
obtenemos 3 ecuaciones que nos dan la solución:
$x = y = z = 4/\lambda$ (un cubo :-)