② a) $w^*_{NN} \geq 0$

$\mathcal{L}_{NN}(w^*_{NN}) = 1/2 \sum_{m=1}^{N} (f_{w^*_{NN}}(x_m) - y_m)^2$

$\mathcal{L}_{LS}(w^*_{LS}) = 1/2 \sum_{m=1}^{N} (w^{*T}_{LS} x_m - y_m)^2$

x^0 $(0,\ldots,0)$ $(0,\ldots,0)$ $\ldots$ $(0,\ldots,0)$ $\hat{y}$ 0

L layers with relu $\rightarrow \min(0, x)$

since $x^0 \geq 0$

$w^*_1, \ldots w^*_{L+1} \geq 0$

$\Rightarrow$ relu$(w_i \cdot l) \geq 0$, with l = relu$(w_1 \ldots w_{i-1} x)$ (weights; corresponding relu)

relu $= \min(0, x)$ but $x \geq 0 \Rightarrow f(x) = \text{relu}(x) = x$

$\Rightarrow f_{w^*_{NN}}(x) = w^{*T}_{L+1}(\ldots \text{relu}(w^{*T}_2(\text{relu}(w^{*T}_1 x))\ldots)) = w^{*T}_{L+1} \ldots w^{*T}_1 x$

$\Rightarrow \mathcal{L}_{NN} = 1/2 \sum_{m=1}^{N} \left[\left(\underbrace{w^{*T}_{L+1} \ldots w^{*T}_1}_{w_{NN}} x - y \right)^2 \right]_m$

$\mathcal{L}_{LS} = 1/2 \sum_{m=1}^{N} (w^T x_m - y_m)^2 =$

successive multiplication of matrices has no further "resolution" (same matrix dimension) than a singular matrix, therefore the optimum solution to the NN loss function would yield the same weight matrix as the standard LS loss function. Thus, Lnn == Lls

b) in this case: $\mathcal{L}_{NN} = 1/2 \sum_{m=1}^{N} \left[\left(w^{*T}_{L+1}(\ldots \text{relu}\, w^{*T}_2(\text{relu}(w^{*T}_1 x))) - y \right)^2 \right]_m$

Considering that the w* for the least squares loss returned a non-negative weight vector, combined with the samples being non-negative, we should assume that the regression targets are also non-negative. . . .

With this in mind, and considering that the feed forward neural network can still attain negative weights in the output and hidden layers, we should expect the feed forward neural network to have more accurate predictions. Moreover, since every hidden layer is coupled with an activation function and, therefore, should be able to extract features not exposed in the raw samples, the only situation where any particular weight would be negative, should be when a extracted feature is detrimental to the prediction. Thus, since the LS loss does not have access to this specific feature it can't compensate resulting in a higher combined loss over all samples.