DLCV 2018 Spring HW3 Report

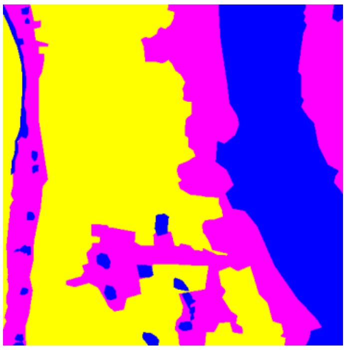
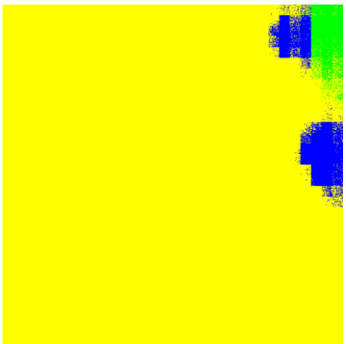
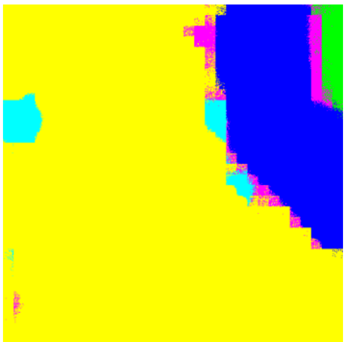
Name: 羅啟心 Dep.: 電機四 Student ID: B03901070

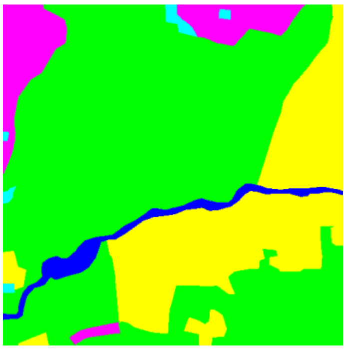
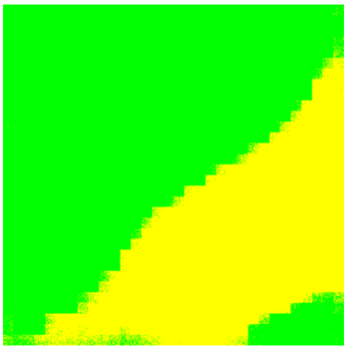
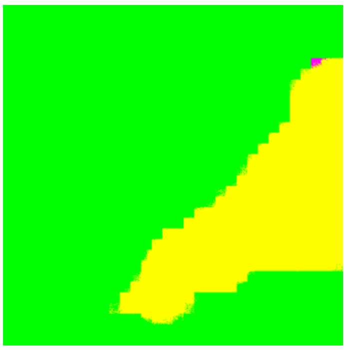
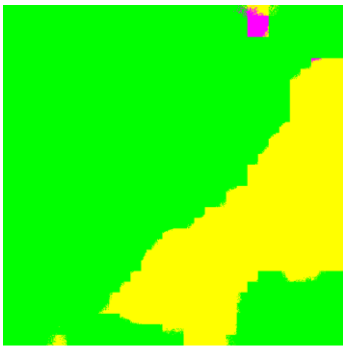
1. (5%) Print the network architecture of your VGG16-FCN32s model.

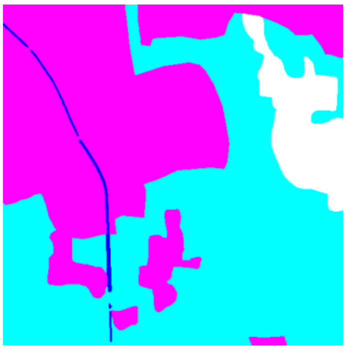
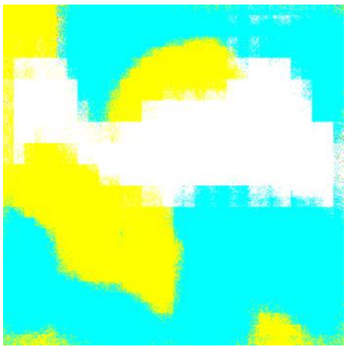
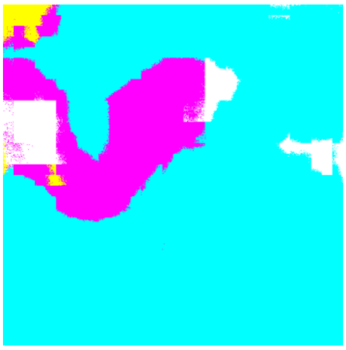
Layer (type)	Output Shape	Param #
input_1 (InputLayer)	(None, 512, 512, 3)	0
block1_conv1 (Conv2D)	(None, 512, 512, 64)	1792
block1_conv2 (Conv2D)	(None, 512, 512, 64)	36928
block1_pool (MaxPooling2D)	(None, 256, 256, 64)	0
block2_conv1 (Conv2D)	(None, 256, 256, 128)	73856
block2_conv2 (Conv2D)	(None, 256, 256, 128)	147584
block2_pool (MaxPooling2D)	(None, 128, 128, 128)	0
block3_conv1 (Conv2D)	(None, 128, 128, 256)	295168
block3_conv2 (Conv2D)	(None, 128, 128, 256)	590080
block3_conv3 (Conv2D)	(None, 128, 128, 256)	590080
block3_pool (MaxPooling2D)	(None, 64, 64, 256)	0
block4_conv1 (Conv2D)	(None, 64, 64, 512)	1180160
block4_conv2 (Conv2D)	(None, 64, 64, 512)	2359808
block4_conv3 (Conv2D)	(None, 64, 64, 512)	2359808
block4_pool (MaxPooling2D)	(None, 32, 32, 512)	0
block5_conv1 (Conv2D)	(None, 32, 32, 512)	2359808
block5_conv2 (Conv2D)	(None, 32, 32, 512)	2359808
block5_conv3 (Conv2D)	(None, 32, 32, 512)	2359808
block5_pool (MaxPooling2D)	(None, 16, 16, 512)	0
fc1 (Conv2D)	(None, 10, 10, 4096)	102764544
fc2 (Conv2D)	(None, 10, 10, 4096)	16781312
conv2d_1 (Conv2D)	(None, 10, 10, 8)	32776
conv2d_transpose_1 (Conv2DTr	(None, 320, 320, 8)	262152

2. (10%) Show the predicted segmentation mask of validation/0008_sat.jpg, validation/0097_sat.jpg, validation/0107_sat.jpg during the early, middle, and the final stage during the training stage. (For example, results of 1st, 10th, 20th epoch)

0008 Ground truth :	0008_sat.jpg	Epoch = 1 (miou = 0.51)

		
Epoch = 5 (miou = 0.593)	Epoch = 10 (miou = 0.622)	Epoch = 50 (miou = 0.658)
		

0097 Ground truth :	0097_sat.jpg	Epoch = 1 (miou = 0.51)
		
Epoch = 5 (miou = 0.593)	Epoch = 10 (miou = 0.622)	Epoch = 50 (miou = 0.658)
		

0107 Ground truth :	0107_sat.jpg	Epoch = 1 (miou = 0.51)
		
Epoch = 5 (miou = 0.593)	Epoch = 10 (miou = 0.622)	Epoch = 50 (miou = 0.658)
		

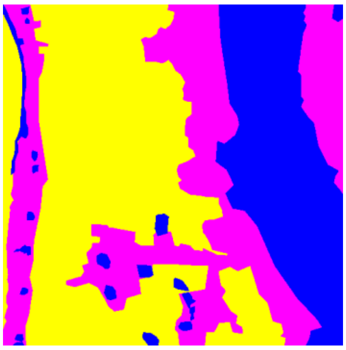
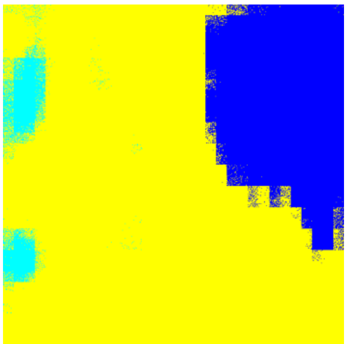
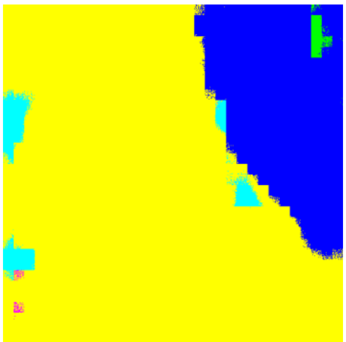
3. (15%) Implement an improved model which performs better than your baseline model. Print the network architecture of this model.

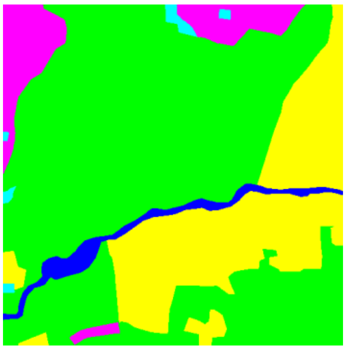
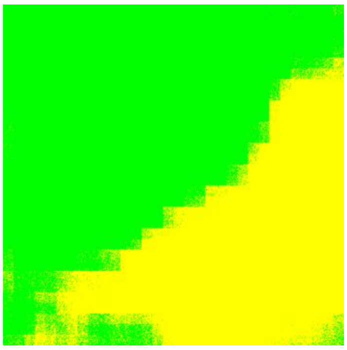
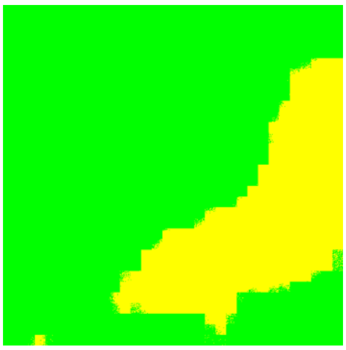
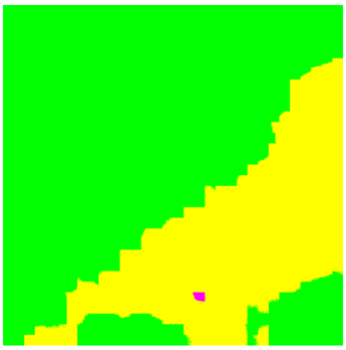
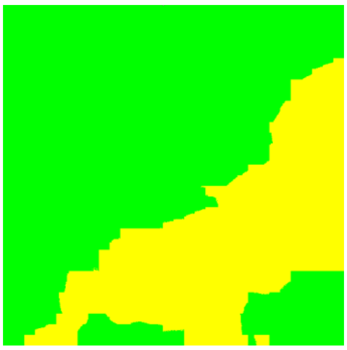
Vgg16(same as in 1.) +

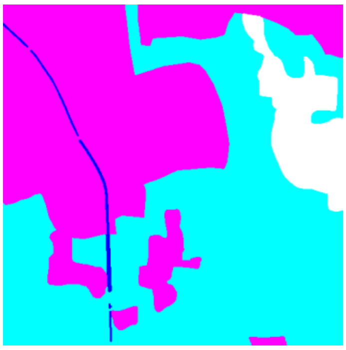
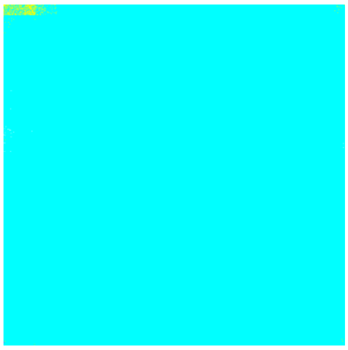
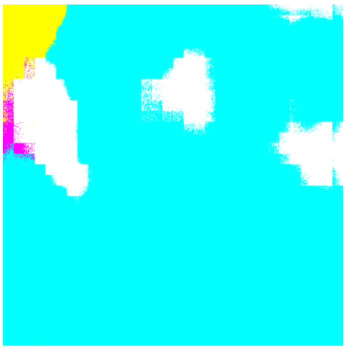
```
=================================================================
ast1 (Conv2D)                (None, 10, 10, 4096)      102764544
_________________________________________________________________
ast2 (Conv2D)                (None, 10, 10, 4096)      16781312
_________________________________________________________________
dropout_1 (Dropout)          (None, 10, 10, 4096)      0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 10, 10, 8)         32776
_________________________________________________________________
conv2d_transpose_1 (Conv2DTr (None, 320, 320, 8)       262152
```

4. (10%) Show the predicted segmentation mask of validation/0008_sat.jpg, validation/0097_sat.jpg, validation/0107_sat.jpg during the early, middle, and the final stage during the training process of this improved model.

0008 Ground truth :	0008_sat.jpg	Epoch = 1 (miou = 0.52)

		
Epoch = 5 (miou = 0.602)	Epoch = 20 (miou = 0.643)	Epoch = 50 (miou = 0.664)
		

0097 Ground truth :	0097_sat.jpg	Epoch = 1 (miou = 0.52)
		
Epoch = 5 (miou = 0.602)	Epoch = 20 (miou = 0.643)	Epoch = 50 (miou = 0.664)
		

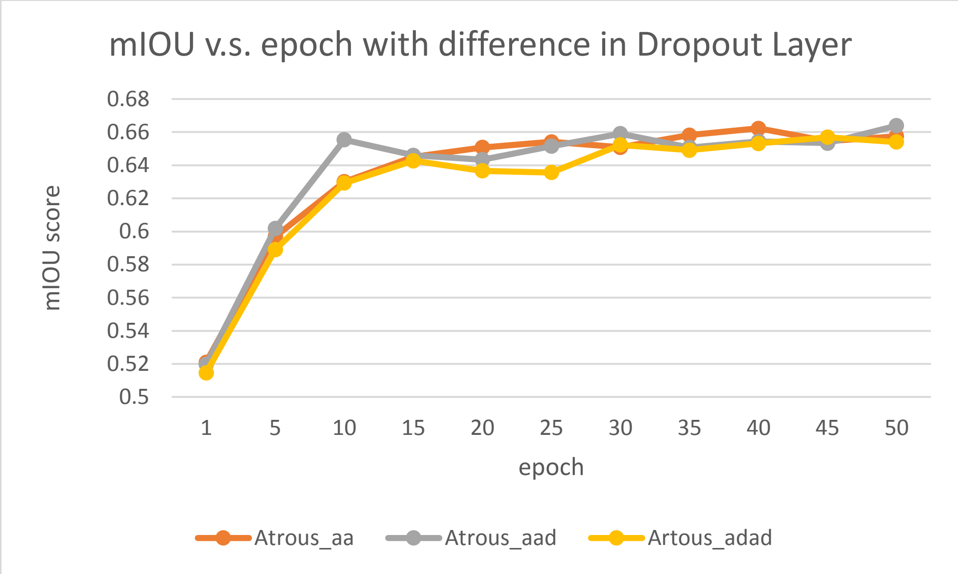
0107 Ground truth :	0107_sat.jpg	Epoch = 1 (miou = 0.52)
		
Epoch = 5 (miou = 0.602)	Epoch = 20 (miou = 0.643)	Epoch = 50 (miou = 0.664)
		

5. (15%) Report mIoU score of both models on the validation set. Discuss the reason why the improved model performs better than the baseline one. You may conduct some experiments and show some evidences to support your discussion.

(a)

Model name	mIOU score
FCN_Vgg16_32s (without dropout, batch size = 4, 50 epochs)	0.658147
Atrous_FCN_Vgg16_32s_aad (with one dropout, batch size = 4,40 epochs)	0.663953

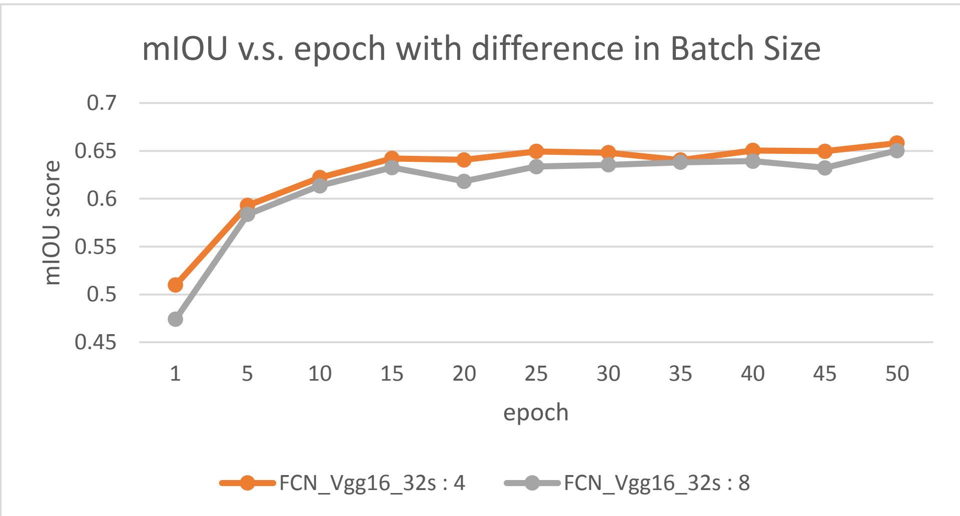
(b) Experiment 1 : Difference in dropout layer



Model	Atrous_aa	Atrous_aad	Atrous_adad
Best mIOU	0.662247	0.663953	0.656973

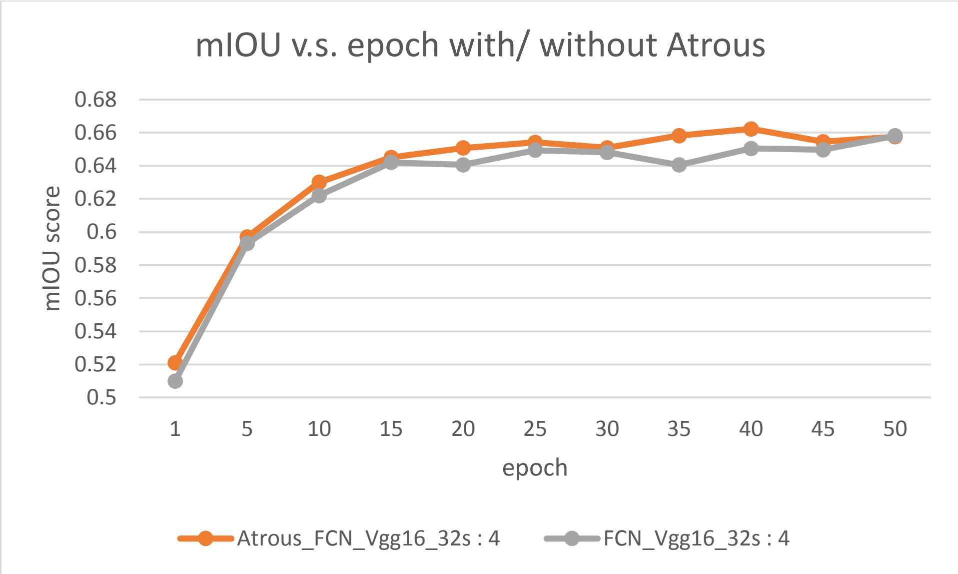
Dropout layer is known to prevent over-fitting. Here, I tried three different arranges of dropout layers, where Atrous_aaa is Vgg16 folowed by two atrous convolution layer; Atrous_aad is Vgg16 followed by two atrous layer, one dropout layer; Atrous_adad is Vgg16 followed by one atrous layer, one dropout layer, another atrous and dropout layer; with all of them later followed by one normal convolution layer and one deconvolution layer. Too much dropout layer may cause model to learn slower.

(c) Experiment 2 : Reduce batch size

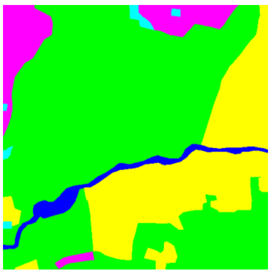
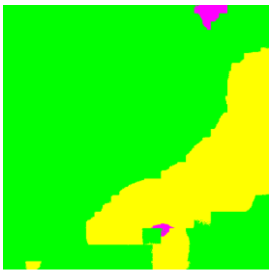


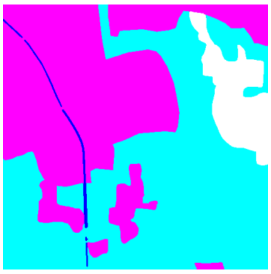
Here, FCN_Vgg16_32s : 4 is with batch size = 4, FCN_Vgg16_32s : 8 is with batch size = 8. We can see that FCN_Vgg16_32s : 4 performs slightly better than FCN_Vgg16_32s : 8, which verifies that small batch size increases the performance in diverse dataset.

(d) Experiment 3: Atrous convolution



Here, Atrous_FCN_Vgg16_32s is with two atrous convolution, one normal convolution and one deconvolution layer at the end, while FCN_Vgg16_32s is with three normal convolution and one deconvolution layer at the end. The plot shows that Atrous_FCN_Vgg16_32s performs slightly better than FCN_Vgg16_32s, and from the figures below, we can see that atrous convolution is able to gather information in wider area, so there are less mixed color.

mask.png	FCN_Vgg16_32s epoch = 50, mIOU = 0.658	Atrous_FCN_Vgg16_32s_aa epoch = 40, mIOU = 0.662
		

6. (5%) [bonus] Calculate the result of d/dw G(w):

objective function:

$$G(\boldsymbol{w}) = -\sum_n \left[t^{(n)} \log \mathrm{x}(\boldsymbol{z}^{(n)}; \boldsymbol{w}) + (1 - t^n) \log \left(1 - \mathrm{x}(\boldsymbol{z}^{(n)}; \boldsymbol{w})\right)\right] \quad \geq 0$$

$$\boldsymbol{w}^* = \arg\min_{\boldsymbol{w}} G(\boldsymbol{w})$$ choose the weights that minimise the network's surprise about the training data

$$\frac{\mathrm{d}}{\mathrm{d}\boldsymbol{w}} G(\boldsymbol{w}) = \sum_n \frac{\mathrm{d}G(\boldsymbol{w})}{\mathrm{d}x^{(n)}} \frac{\mathrm{d}x^{(n)}}{\mathrm{d}\boldsymbol{w}} = -\sum_n (t^{(n)} - x^{(n)}) \boldsymbol{z}^{(n)}$$ = prediction error x feature

$$\boldsymbol{w} \leftarrow \boldsymbol{w} - \eta \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{w}} G(\boldsymbol{w})$$ iteratively step down the objective (gradient points up hill) 39

Ans :

The loss function of logistic regression is :

$$G(\omega) = -\sum_{n=1}^{N} [t^{(n)} \times \left(log\left(x\left(\omega^T \times z^{(n)}\right)\right)\right) + (1 - t^{(n)}) \times (log(1 - x(\omega^T \times z^{(n)})))]$$

And we aim to obtain is derivative

(1) First, By chain rule :

$$\frac{dG_n(\omega)}{d\omega_j} = \sum_{n=1}^{N} \frac{dG_n(\omega)}{dx^{(n)}} \times \frac{dx^{(n)}}{d\omega_j}$$

$here,$

$$G_n(\omega) = -[t^{(n)}\left(log\left(x\left(\omega^T z^{(n)}\right)\right)\right) + \left(1 - t^{(n)}\right)\left(\log\left(1 - x\left(\omega^T z^{(n)}\right)\right)\right)],$$

$$x^{(n)} = x(\omega^T z^{(n)})$$

(2) The first term in (1) is :

$$\frac{dG_n(\omega)}{dx^{(n)}} = \frac{t^{(n)}}{x^{(n)}} - \frac{1 - t^{(n)}}{1 - x^{(n)}}$$

(3) The second term in (1) is :

$$\frac{dx^{(n)}}{d\omega_j} = \frac{dx_n}{d\omega^T z^{(n)}} \frac{d\omega^T z^{(n)}}{d\omega_j} = x^{(n)}(1 - x^{(n)}) z^{(n,j)}$$

$$Since\ (assumption\ of\ logistic\ regression)\ \frac{dx(\alpha)}{d\alpha} = x(\alpha)(1 - x(\alpha))$$

$$here,\ \ z^{(n,i)}\ is\ the\ i\ th\ element\ of\ z^{(n)}$$

(4)

$$\frac{dG_n(\omega)}{d\omega_j} = \sum_{i=1}^{n} \frac{dG_n(\omega)}{dx^{(n)}} \times \frac{dx^{(n)}}{d\omega_j} = \left(\frac{t^{(n)}}{x^{(n)}} - \frac{1-t^{(n)}}{1-x^{(n)}}\right) x^{(n)}\left(1-x^{(n)}\right) z^{(n,j)}$$

$$= \left(t^{(n)}\left(1-x^{(n)}\right) - \left(1-t^{(n)}\right)x^{(n)}\right) z^{(n,j)}$$

$$= \left(t^{(n)} - t^{(n)}x^{(n)} - x^{(n)} + t^{(n)}x^{(n)}\right) z^{(n,j)}$$

$$= (t^{(n)} - x^{(n)}) z^{(n,j)}$$

Hence, $\frac{dG(\omega)}{d\omega_j} = \sum_{n=1}^{N} (t^{(n)} - x_n) z^{(n,j)}$

That is, $\frac{dG(\omega)}{d\omega} = \sum_{n=1}^{N} (t^{(n)} - x_n) z^{(n)}$

References :

1. https://stats.stackexchange.com/questions/278771/how-is-the-cost-function-from-logistic-regression-derivated
2. http://www.cs.columbia.edu/~blei/fogm/2014F/lectures/glms.pdf