Karan Agarwalla – 180050045

Problem 1.1(b)

There are four features for counts of pixels with 1, 2, 3 and 4 black neighbors. The 5th feature is the bias term (1). The features are normalized except for the fifth feature. In the implementation the fifth feature is the first and the other four follows. Accuracy observed is 100%.
The features designed must be independent of orientation and translation of figure. Hence features dependent on means and standard deviation of x-coordinate and y-coordinate that I tried didn't give high accuracy. The counts are independent of the orientation hence, give high accuracy. The combination of these features can be used to represent other metrics like area (nearly the summation of the four features).

Problem 2

Problem 2.1

(a)

$$P\big(Y=1\big|w_1,\phi(x)\big)=\frac{e^{w_1^T\phi(x)}}{e^{w_0^T\phi(x)}+e^{w_1^T\phi(x)}}=\frac{1}{1+\ e^{-(w_1-w_0)^T\phi(x)}}=\sigma_w(x))$$

$$P\big(Y=0\big|w_0,\phi(x)\big)=\frac{e^{w_0^T\phi(x)}}{e^{w_0^T\phi(x)}+e^{w_1^T\phi(x)}}=\frac{e^{-(w_1-w_0)^T\phi(x)}}{1+\ e^{-(w_1-w_0)^T\phi(x)}}=1-\ \sigma_w(x))$$

Hence sigmoid probability in case of binary logistic regression is a special case of multi-class regression with weight vector $w = w_1 – w_0$. Also $y^{(i)}$ in case of logistic regression is simply $y_1^{(i)}$ and $y_0^{(i)} = 1 - y_1^{(i)}$. Hence, simplifying the expression,

$$E(W)=\ -\frac{1}{N}\sum_{i=1}^{N}\sum_{k=0}^{1}y_k^{(i)}*\log\left(P\left(Y=k\middle|w_k,\phi\big(x^{(i)}\big)\right)\right)$$

$$=\ -\frac{1}{N}\sum_{i=1}^{N}y^{(i)}\log\left(\sigma_w\big(x^{(i)}\big)\right)+(1-y^{(i)})\log\left(1-\sigma_w\big(x^{(i)}\big)\right)$$

which is the cross-entropy loss function used to train binary logistic regression.

(b)

$$\sum_{k=1}^{K}y_k^{(i)}=1\ \forall\ i\in[1,N] \qquad (1)$$

$$P\left(Y=k\middle|w_k,\phi\big(x^{(i)}\big)\right)=\frac{e^{w_k^T\phi\left(x^{(i)}\right)}}{\sum_{l=1}^{K}e^{w_l^T\phi\left(x^{(i)}\right)}}$$

$$\log\left(P\left(Y=k\middle|w_k,\phi\big(x^{(i)}\big)\right)\right)=\log\left(\frac{e^{w_k^T\phi\left(x^{(i)}\right)}}{\sum_{l=1}^{K}e^{w_l^T\phi\left(x^{(i)}\right)}}\right)=w_k^T\phi\big(x^{(i)}\big)-\log\left(\sum_{l=1}^{K}e^{w_l^T\phi\left(x^{(i)}\right)}\right)$$

$$E(W)=\ -\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K}y_k^{(i)}*\log\left(P\left(Y=k\middle|w_k,\phi\big(x^{(i)}\big)\right)\right)$$

$$= -\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K} y_k^{(i)} * \log\left(P\left(Y = k\middle|w_k, \phi\left(x^{(i)}\right)\right)\right)$$

$$= -\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K} y_k^{(i)} * \left[w_k^T\phi\left(x^{(i)}\right) - \log\left(\sum_{l=1}^{K} e^{w_l^T\phi\left(x^{(i)}\right)}\right)\right]$$

$$= -\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K} y_k^{(i)} * \left[w_k^T\phi\left(x^{(i)}\right)\right] + \frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K} y_k^{(i)} \log\left(\sum_{l=1}^{K} e^{w_l^T\phi\left(x^{(i)}\right)}\right)$$

$$= -\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K} y_k^{(i)} * \left[w_k^T\phi\left(x^{(i)}\right)\right] + \frac{1}{N}\sum_{i=1}^{N} \log\left(\sum_{l=1}^{K} e^{w_l^T\phi\left(x^{(i)}\right)}\right)$$

(follows from (1))

$$\frac{\partial(E(W))}{\partial w_j} = \frac{\partial}{\partial w_j}\left[-\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K} y_k^{(i)} * \left[w_k^T\phi\left(x^{(i)}\right)\right] + \frac{1}{N}\sum_{i=1}^{N} \log\left(\sum_{l=1}^{K} e^{w_l^T\phi\left(x^{(i)}\right)}\right)\right]$$

$$= \frac{\partial}{\partial w_j}\left[-\frac{1}{N}\sum_{i=1}^{N}\left[y_j^{(i)} w_j^T\phi\left(x^{(i)}\right)\right] + \frac{1}{N}\sum_{i=1}^{N} \log\left(\sum_{l=1}^{K} e^{w_l^T\phi\left(x^{(i)}\right)}\right)\right]$$

$$= -\frac{1}{N}\left[\sum_{i=1}^{N} \mathrm{y_j^{(i)}}\phi(\mathrm{x^{(i)}})\right] + \frac{1}{N}\left[\sum_{i=1}^{N} \frac{e^{w_j^T\phi\left(x^{(i)}\right)} * \phi\left(x^{(i)}\right)}{\sum_{l=1}^{K} e^{w_l^T\phi\left(x^{(i)}\right)}}\right]$$

$$= -\frac{1}{N}\left[\sum_{i=1}^{N} \mathrm{y_j^{(i)}}\phi(\mathrm{x^{(i)}})\right) - \sum_{i=1}^{N} P\left(Y = j\middle|w_j, \phi\left(x^{(i)}\right)\right) * \phi\left(x^{(i)}\right)]$$

Let's define $P_{ik} = P\left(Y = k\middle|w_k, \phi\left(x^{(i)}\right)\right)$, i.e., probability of kth label on ith example

$$= -\frac{1}{N}\left[\sum_{i=1}^{N} (\mathrm{y_j^{(i)}} - P_{ij})\phi(\mathrm{x^{(i)}})\right]$$

Hence, $\frac{\partial(E(W))}{\partial w_j} = = \frac{1}{N}\left[\sum_{i=1}^{N}(P_{ij} - \mathrm{y_j^{(i)}})\phi(\mathrm{x^{(i)}})\right] = \frac{1}{N}\phi(X)^T\left(P_j - Y_j\right)$

where P_j is vector of size N*1 whose entries are P_{ij} for each example i. Similarly, Y_j is vector of size N*1 whose entries are $Y_i^{(j)}$ for each example

Hence, $\frac{\partial(E(W))}{\partial W} = \frac{1}{N}\phi(X)^T(P - Y)$

where P and Y are matrices of size N*K where K is number of classes and $\phi(X)$ is matrix of size N*F where F is number of features.

Problem 2.2

(b)

Test accuracy obtained = 86.32%

Test accuracy for model M = 84.18%

Accuracy is not a good metric when the number of test examples for a particular class dominate the dataset (here class 0). Since in such a case the accuracy of that particular class dominates the evaluation of the model. Here in model M we always predict 0, hence we achieve quite high accuracy score. Hence we need a metric that specifies how well the model does on each class taking into account both false negatives and false positives.

(c)

F1 score obtained for test set: 0.301
F1 score for model: 0

F1 is a good evaluation metric since it strikes a balance between precision and recall. It takes into account misclassifications for each class and is not dependent on the majority class (here 0). F scores evaluate how well the model does on each class.

(e)

Test accuracy for logistic regression: 84.435%
Test accuracy for perceptron: 78.28%

Hence logistic regression achieves higher accuracy. This can be attributed to the smooth differentiable objective function in case of logistic regression against a step function in perception. Hence it is better able to converge to minima and is softer than perceptron, hence finding a better decision boundary.