Devon Welch, 999171747

1.1

$$p(y=1|x) = \frac{p(y=1)p(x|y=1)}{p(x)}$$

Using Baye's Rule:

$$= \frac{\propto \left(\prod_{i=1}^{D}\left(\frac{1}{\sigma_i\sqrt{2\pi}}\right)\exp\left(-\frac{(x_i-\mu_{i1})^2}{2\sigma_i^2}\right)\right)}{\left(\propto \prod_{i=1}^{D}\left(\frac{1}{\sigma_i\sqrt{2\pi}}\right)\exp\left(-\frac{(x_i-\mu_{i1})^2}{2\sigma_i^2}\right)\right) + \left((1-\propto)\prod_{i=1}^{D}\left({}^{1}/_{\sigma_i\sqrt{2\pi}}\right)\exp\left(-\frac{(x_i-\mu_{i0})^2}{2\sigma_i^2}\right)\right)}$$

Simplifying by $p(x|y=1)$:

$$= \frac{\propto}{\propto + (1-\propto)\prod_{i=1}^{D}\left({}^{1}/_{\sigma_i\sqrt{2\pi}}\right)\exp\left(-\frac{(x_i-\mu_{i0})^2}{2\sigma_i^2}\right) / \left(\prod_{i=1}^{D}\left(\frac{1}{\sigma_i\sqrt{2\pi}}\right)\exp\left(-\frac{(x_i-\mu_{i1})^2}{2\sigma_i^2}\right)\right)}$$

Simplifying by $\propto$:

$$= \frac{1}{1 + (1-\propto)\prod_{i=1}^{D}\left({}^{1}/_{\sigma_i\sqrt{2\pi}}\right)\exp\left(-\frac{(x_i-\mu_{i0})^2}{2\sigma_i^2}\right) / \left(\propto \prod_{i=1}^{D}\left(\frac{1}{\sigma_i\sqrt{2\pi}}\right)\exp\left(-\frac{(x_i-\mu_{i1})^2}{2\sigma_i^2}\right)\right)}$$

$\left(\frac{1}{\sigma_i\sqrt{2\pi}}\right)$ can be cancelled in the denominator, and $\frac{a^x}{a^y} = a^{x-y}$

$$= \frac{1}{1 + \left({}^{(1-\propto)}/_{\propto}\right)\exp\left(\sum_{i=1}^{D} -\frac{(x_i-\mu_{i0})^2}{2\sigma_i^2} + \frac{(x_i-\mu_{i1})^2}{2\sigma_i^2}\right)}$$

Combining fractions

$$= \frac{1}{1 + \left({}^{(1-\propto)}/_{\propto}\right)\exp\left(\sum_{i=1}^{D} \frac{-2x_i\mu_{i1} + \mu_{i1}^2 + 2x_i\mu_{i0} - \mu_{i0}^2}{2\sigma_i^2}\right)}$$

Changing $\left({}^{(1-\propto)}/_{\propto}\right)$ into a form that will allow it o be put inside the exponential

$$= \frac{1}{1 + \exp(\ln\left(\frac{(1-\propto)}{\propto}\right)) \exp\left(\sum_{i=1}^{D} \frac{-2x_i\mu_{i1} + \mu_{i1}^2 + 2x_i\mu_{i0} - \mu_{i0}^2}{2\sigma_i^2}\right)}$$

$$= \frac{1}{1 + \exp\left(\ln\left(\frac{(1-\propto)}{\propto}\right) + \left(\sum_{i=1}^{D} \frac{-2x_i\mu_{i1} + \mu_{i1}^2 + 2x_i\mu_{i0} - \mu_{i0}^2}{2\sigma_i^2}\right)\right)}$$

Isolating x values

$$= \frac{1}{1 + \exp\left(\sum_{i=1}^{D} \frac{-2x_i\mu_{i1} + 2x_i\mu_{i0}}{2\sigma_i^2} + \frac{\mu_{i1}^2 - \mu_{i0}^2}{2\sigma_i^2} + \ln\left(\frac{(1-\propto)}{\propto}\right)\right)}$$

$$= \frac{1}{1 + \exp\left(\sum_{i=1}^{D} \frac{-x_i\mu_{i1} + x_i\mu_{i0}}{\sigma_i^2} + \sum_{i=1}^{D} \frac{\mu_{i1}^2 - \mu_{i0}^2}{2\sigma_i^2} + \ln\left(\frac{(1-\propto)}{\propto}\right)\right)}$$

$$= \frac{1}{1 + \exp\left(\sum_{i=1}^{D} x_i((\mu_{i0} - \mu_{i1})/\sigma_i^2) + \sum_{i=1}^{D} \frac{\mu_{i1}^2 - \mu_{i0}^2}{2\sigma_i^2} + \ln\left(\frac{(1-\propto)}{\propto}\right)\right)}$$

Converting to desired form (μ signs are switched to accommodate).

$$= \frac{1}{1 + \exp\left(-\sum_{i=1}^{D} x_i((\mu_{i1} - \mu_{i0})/\sigma_i^2) - \left(\left(\sum_{i=1}^{D} \frac{\mu_{i0}^2 - \mu_{i1}^2}{2\sigma_i^2}\right) - \ln(\propto/(1-\propto)\right)\right)}$$

$$w = (\mu_{i1} - \mu_{i0})/\sigma_i^2$$

$$b = \left(\left(\sum_{i=1}^{D} \frac{\mu_{i0}^2 - \mu_{i1}^2}{2\sigma_i^2}\right) - \ln(\propto/(1-\propto)\right)$$

1.2

Finding $p(y = 0|x)$

$$p(y=0|x) = 1 - \frac{1}{1+\exp\left(-\sum_{i=1}^{D} x_i((\mu_{i1}-\mu_{i0})/\sigma_i^2) - \left(\left(\sum_{i=1}^{D}\frac{\mu_{i0}^2-\mu_{i1}^2}{2\sigma_i^2}\right) + \ln(\propto/(1-\propto)\right)\right)}$$

$$= \frac{\exp\left(-\sum_{i=1}^{D} x_i((\mu_{i1}-\mu_{i0})/\sigma_i^2) - \left(\left(\sum_{i=1}^{D}\frac{\mu_{i0}^2-\mu_{i1}^2}{2\sigma_i^2}\right) + \ln(\propto/(1-\propto)\right)\right)}{1+\exp\left(-\sum_{i=1}^{D} x_i((\mu_{i1}-\mu_{i0})/\sigma_i^2) - \left(\left(\sum_{i=1}^{D}\frac{\mu_{i0}^2-\mu_{i1}^2}{2\sigma_i^2}\right) + \ln(\propto/(1-\propto)\right)\right)}$$

Finding $E(w,b)$

$$p\left(y^{(1)},\dots\ y^{(N)}\middle|x^{(1)},\dots\ x^{(N)},w,b\right) = \prod_{n=1}^{N} p(y^{(n)}|x^{(n)})$$

$$= \prod_{n=1}^{N} p\left(y^{(n)}=1\middle|x^{(n)}\right)^{y^{(n)}} p(y^{(n)}=0|x^{(n)})^{1-y^{(n)}}$$

Subbing in values of $p\left(y^{(n)}=0\middle|x^{(n)}\right) and\ p(y^{(n)}=1|x^{(n)})$

$$= \prod_{n=1}^{N}\left(\frac{1}{1+\exp(-\sum_{i=1}^{D} w_i {x_i}^{(n)} - b^{(n)})}\right)^{y^{(n)}}\left(\frac{\exp(-\sum_{i=1}^{D} w_i {x_i}^{(n)} - b^{(n)})}{1+\exp(-\sum_{i=1}^{D} w_i {x_i}^{(n)} - b^{(n)})}\right)^{1-y^{(n)}}$$

Taking ln and negating

$$-\ln\left(p\left(y^{(1)},\dots\ y^{(N)}\middle|x^{(1)},\dots\ x^{(N)},w,b\right)\right)$$
$$= -\sum_{n=1}^{N} y^{(n)} ln\left(\frac{1}{1+\exp(-\sum_{i=1}^{D} w_i {x_i}^{(n)} - b^{(n)})}\right)$$
$$-\sum_{n=1}^{N}\left(1-y^{(n)}\right) ln\left(\frac{\exp(-\sum_{i=1}^{D} w_i {x_i}^{(n)} - b^{(n)})}{1+\exp(-\sum_{i=1}^{D} w_i {x_i}^{(n)} - b^{(n)})}\right)$$

Simplifying using the log rule that $\log\left(\frac{1}{x}\right) = -\log(x)$

$$= \sum_{n=1}^{N} y^{(n)} ln\left(1+\exp\left(-\sum_{i=1}^{D} w_i {x_i}^{(n)} - b^{(n)}\right)\right)$$
$$+\sum_{n=1}^{N}\left(1-y^{(n)}\right) ln\left(\frac{1+\exp(-\sum_{i=1}^{D} w_i {x_i}^{(n)} - b^{(n)})}{\exp(-\sum_{i=1}^{D} w_i {x_i}^{(n)} - b^{(n)})}\right)$$

Simplifying using the log rule that $\log\left(\frac{x}{y}\right) = \log(x) - \log(y)$

$$= \sum_{n=1}^{N} y^{(n)} ln\left(1 + \exp\left(-\sum_{i=1}^{D} w_i {x_i}^{(n)} - b^{(n)}\right)\right) + \sum_{n=1}^{N} \left(1 - y^{(n)}\right) ln\left(1 + \exp(-\sum_{i=1}^{D} w_i {x_i}^{(n)} - b^{(n)})\right) - \sum_{n=1}^{N} \left(1 - y^{(n)}\right) ln\left(\exp(-\sum_{i=1}^{D} w_i {x_i}^{(n)} - b^{(n)})\right)$$

Simplifying by grouping brackets

$$E(w,b) = \sum_{n=1}^{N} ln\left(1 + \exp\left(-\sum_{i=1}^{D} w_i {x_i}^{(n)} - b^{(n)}\right)\right) + \sum_{n=1}^{N} \left(1 - y^{(n)}\right)\left(\sum_{i=1}^{D} w_i {x_i}^{(n)} + b^{(n)}\right)$$

Using chain rule

$$\frac{\partial E}{\partial w_j} = \sum_{n=1}^{N} \frac{-x_j^{(n)} \exp\left(-\sum_{i=1}^{D} w_i {x_i}^{(n)} - b^{(n)}\right)}{1 + \exp(-\sum_{i=1}^{D} w_i {x_i}^{(n)} - b^{(n)})} + \sum_{n=1}^{N} \left(1 - y^{(n)}\right) x_j^{(n)}$$

$$\frac{\partial E}{\partial b} = \sum_{n=1}^{N} \frac{-\exp\left(-\sum_{i=1}^{D} w_i {x_i}^{(n)} - b^{(n)}\right)}{1 + \exp(-\sum_{i=1}^{D} w_i {x_i}^{(n)} - b^{(n)})} + \sum_{n=1}^{N} \left(1 - y^{(n)}\right)$$

1.3

$$p\left(y^{(1)}, \ldots\, y^{(N)} \middle| x^{(1)}, \ldots\, x^{(N)}, w, b\right) = \prod_{n=1}^{N} p\left(y^{(n)} \middle| x^{(n)}\right)$$

$$p(w, b|y, x) \propto p(y|x, w, b) p(w) p(b)$$

$$p(w,b|y,x) \propto p(y|x,w,b)N(w|0,\frac{1}{\lambda})N(b|0,\frac{1}{\lambda})$$

$$p(w,b|y,x) \propto \left(\prod_{n=1}^{N} p(y^{(n)}|x^{(n)})\right) N(w|0,\frac{1}{\lambda})N(b|0,\frac{1}{\lambda})$$

$$p(w,b|y,x) \propto \left(\prod_{n=1}^{N} p(y^{(n)}|x^{(n)})\right)\frac{1}{\left(\frac{1}{\sqrt{\lambda}}\right)\sqrt{2\pi}}\exp\left(-\frac{w^2}{2/\lambda}\right)\frac{1}{\left(\frac{1}{\sqrt{\lambda}}\right)\sqrt{2\pi}}\exp\left(-\frac{b^2}{2/\lambda}\right)$$

$$p(w,b|y,x) \propto \left(\prod_{n=1}^{N} p(y^{(n)}|x^{(n)})\right)\frac{\lambda}{2\pi}\exp\left(-\frac{w^2}{2/\lambda}-\frac{b^2}{2/\lambda}\right)$$

$$L(w,b) = E(w,b) - ln\left(\frac{\lambda}{2\pi}\right)+\frac{w^2}{2/\lambda}+\frac{b^2}{2/\lambda}$$

$$L(w,b) = E(w,b) - ln\left(\frac{\lambda}{2\pi}\right)+\frac{\lambda w^2}{2}+\frac{\lambda b^2}{2}$$

$$\frac{\partial L}{\partial w_j} = \sum_{n=1}^{N}\frac{-x_j^{(n)}\exp\left(-\sum_{i=1}^{D} w_i {x_i}^{(n)} - b^{(n)}\right)}{1+\exp(-\sum_{i=1}^{D} w_i {x_i}^{(n)} - b^{(n)})} + \sum_{n=1}^{N}\left(1-y^{(n)}\right)x_j^{(n)} + \lambda w$$

$$\frac{\partial L}{\partial b} = \sum_{n=1}^{N}\frac{-\exp\left(-\sum_{i=1}^{D} w_i {x_i}^{(n)} - b^{(n)}\right)}{1+\exp(-\sum_{i=1}^{D} w_i {x_i}^{(n)} - b^{(n)})} + \sum_{n=1}^{N}\left(1-y^{(n)}\right) + \lambda b$$

2.1

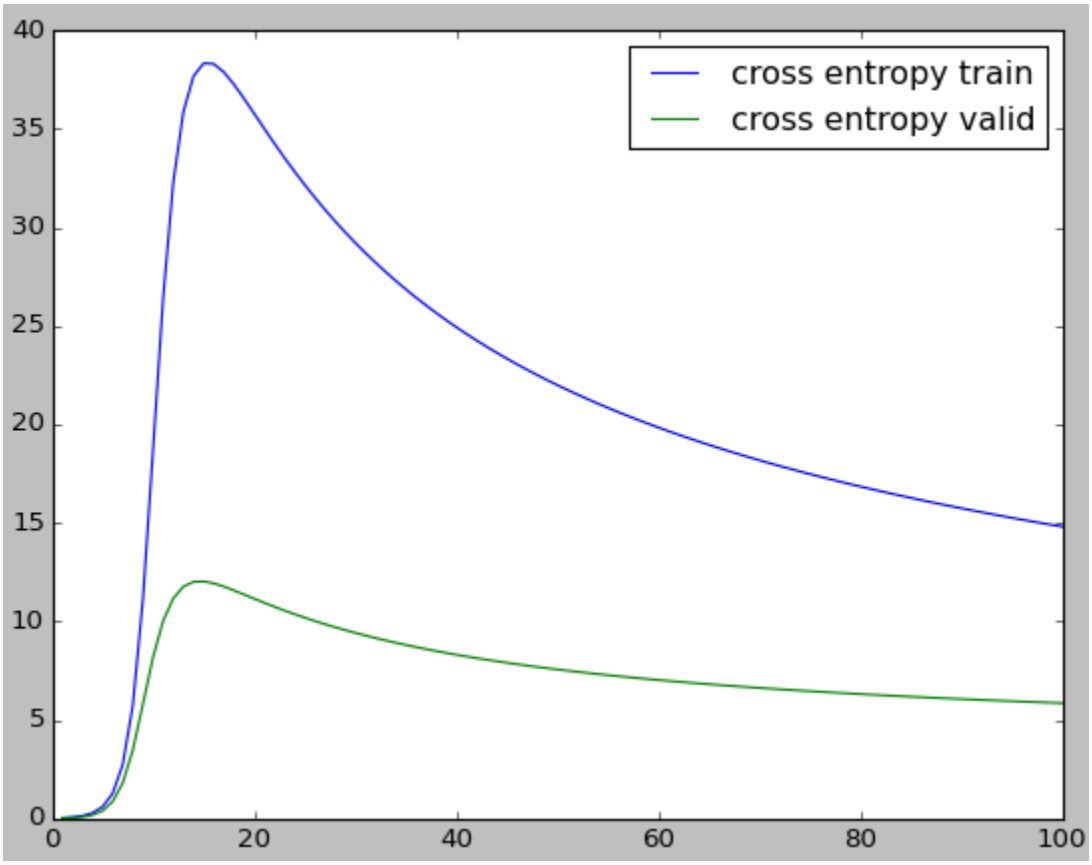
The training classification rates for 1, 3, 5, 7, and 9 are 0.82, .086, 0.86, 0.86, and 0.84, respectively. I would choose 5 as the best value of k, as it's tied for the highest classification rate, and is in the middle so it is farther form the dropoff at both ends. K+2 (7) and K-2 (3) both have the same rate of 0.86. The test classification rate for all of these three is higher, at .92 for 3 and .94 for 5 and 7.

2.2

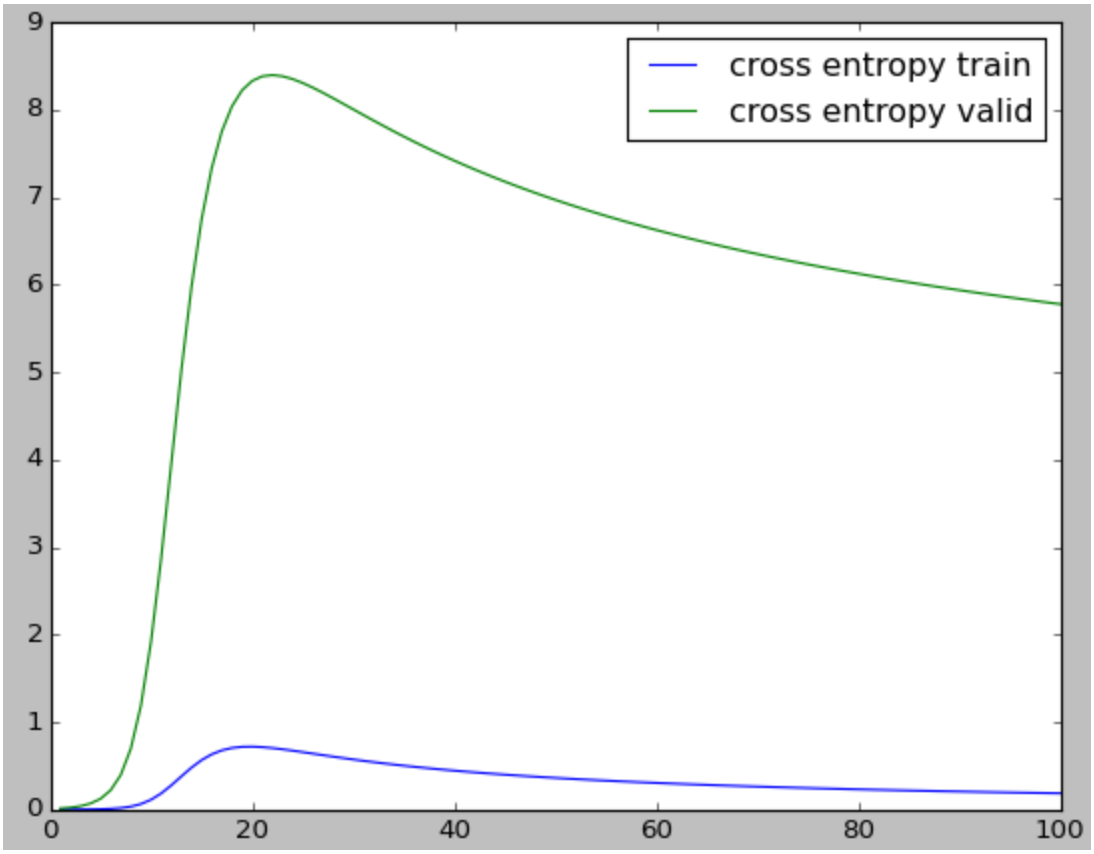
The hyperparameters which I determined to be best were a learning rate of 0.05, and 100 iterations. The final cross entropy and classification error for train, valid and test, and mnist_train and mnist_trian_small respectively, are: mnist_train: (train cross entropy: 14.811450, train classification error: 0.0373, valid cross entropy: 5.856860, valid classification error: 0.0980, test cross entropy: 4.85316136724, test classification error: 0.0785), mnist_train_small: (train cross

entropy: 0.186111, train classification error: 0.0909, valid cross entropy: 5.778267, valid classification error: 0.3529, test cross entropy: 4.16516872097, test classification error: 0.2942)

mnist_train:



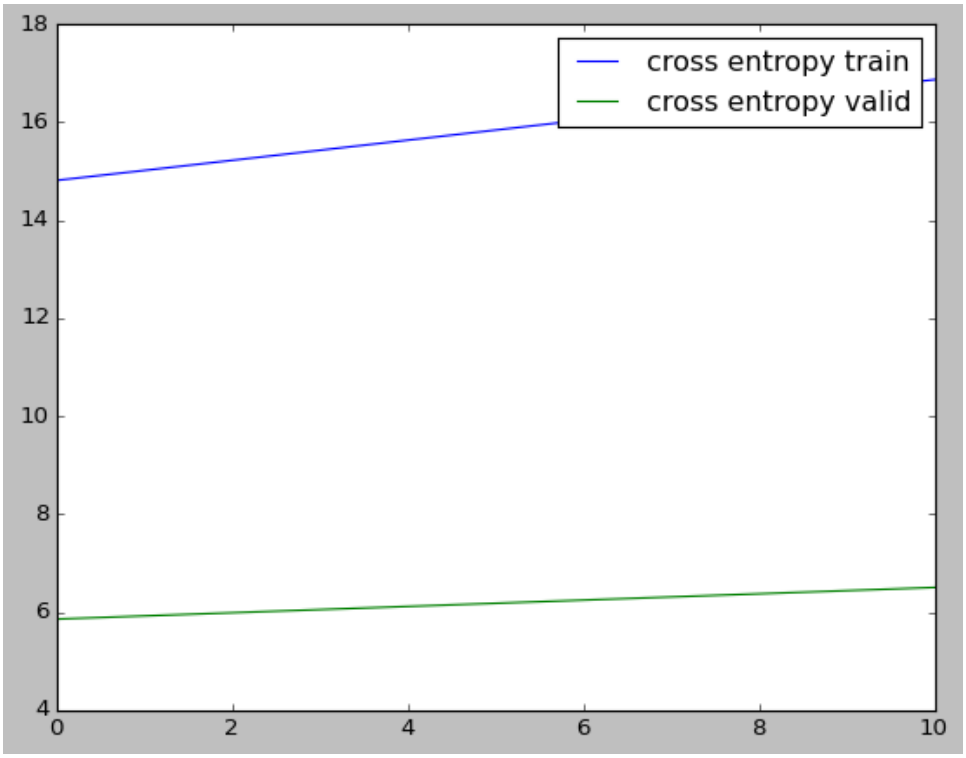

mnist_train_small:

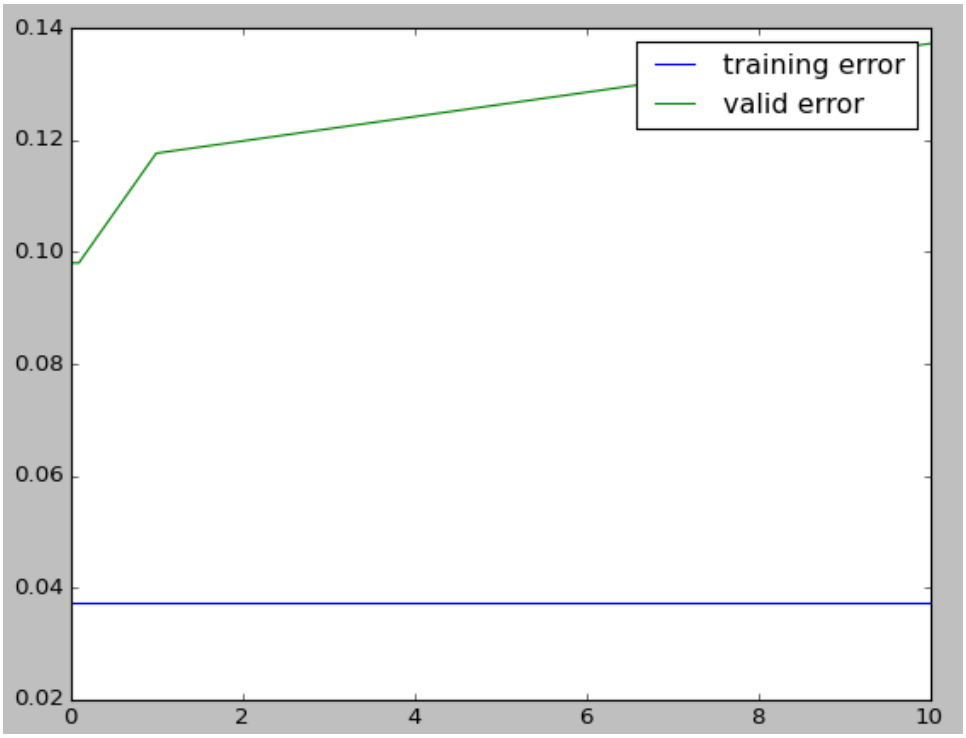



The results don't change on different runs because I initialize my weights to set, relatively low numbers (0.1). I chose static weights rather than random weights because they reliably provided better classification rates than any randomly generated weights, and generally with fewer iterations, too.

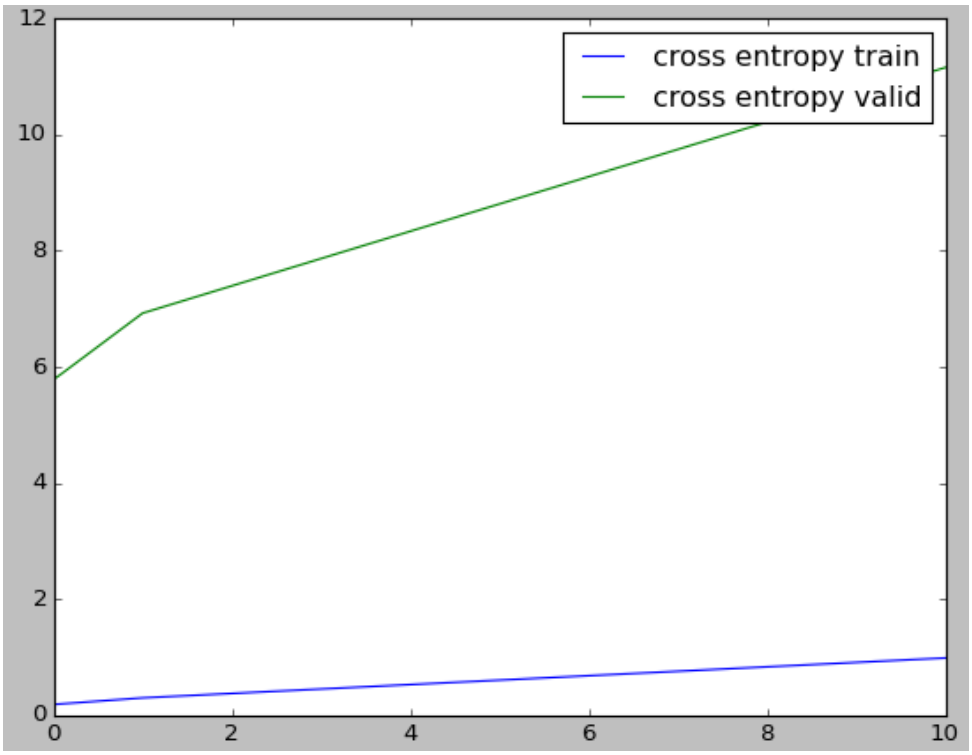
2.3

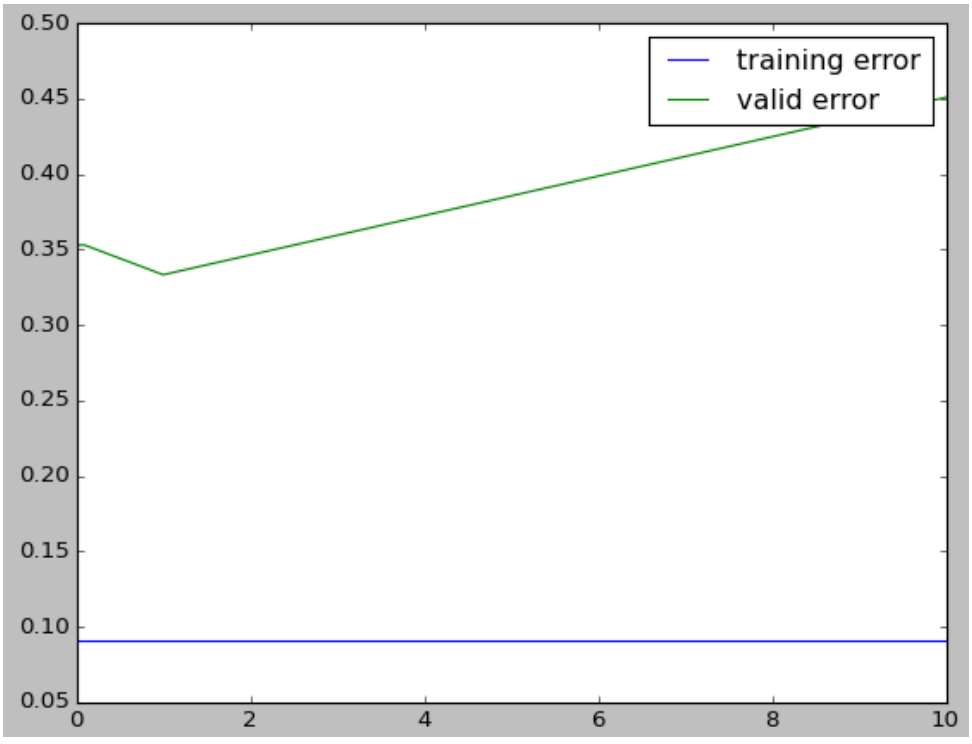
mnist_train:




mnist_train_small:



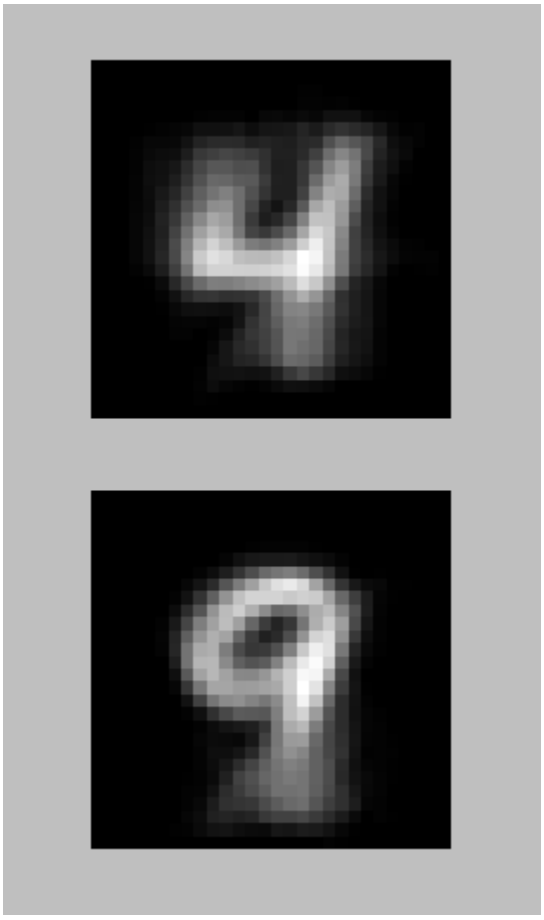

Cross entropy and classification error increase as λ increases, with one exception in valid_error when using the small training set. This is probably because the model is overfitting, and the values of λ to the left of the minimum are too small to prevent the overfitting, whereas the values to the right hinder the fitting in the first place. So, the best value of λ is probably based on the

size of your dataset; as can be seen in the graphs above, valid error decreases to the left of $\lambda = 1$ in the large training set, but increases in the small dataset; therefore, there is not a catch-all value that is the best, but in this case, I would have to say that the better value is the one that works for larger datasets, meaning $\lambda = 0.01$. The test error for the large dataset is 0.07843137254901966 and 0.29411764705882343 for the small set.

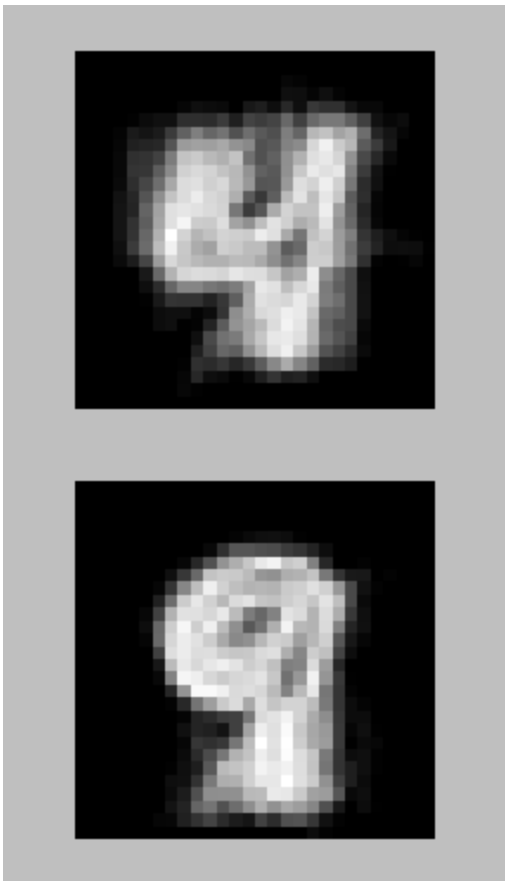
2.4

The train accuracy is 0.8625, and the test accuracy is 0.8

Mean:



Var:



The bright spots in the mean and the dark spots in the var indicate where the most reliability is. The var is quite fuzzy overall, indicating that there is a lot of uncertainty.

2.5

Logistic regression has the best classification rate, followed by k-NN, followed by naïve Bayes. I expected Bayes to have the worst classification rate because fours and nines are very similar – if the digit classification had been on classes that were more drastically different, I think that Bayes would have performed better. I think this because Bayes generates models of what it thinks its classes are, and so the more similar the classes, the less easily it can discriminate between them. And on the other side, I think that it makes sense that logistic regression has the best classification rate, because it is iterative and so it can tune itself to the data better. Lastly, I think that it makes sense that kNN is in the middle, because it simply find which classes are closest to any new data point.