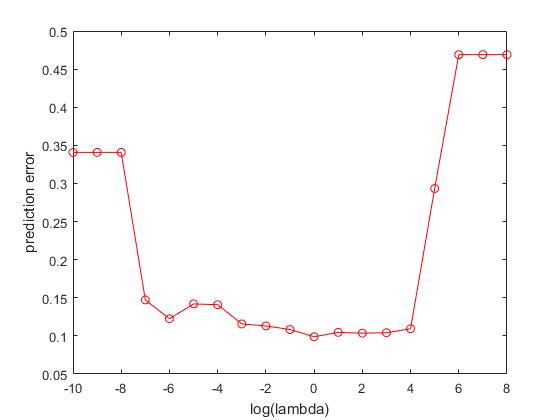
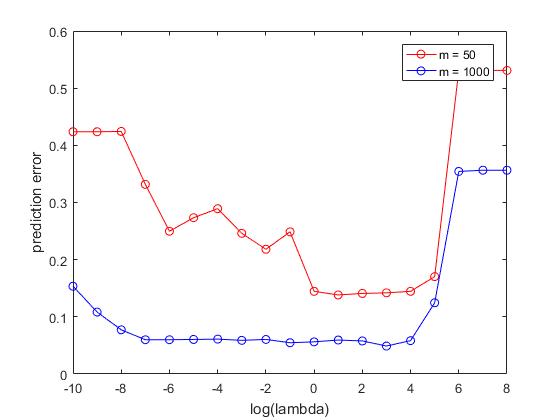


3. a.



3. b.



3. c. Similarities: In both lines, the error is relatively low when 0 <= log(lambda) <= 4

Differences: When m is 50, the error is high when lambda is very small or very large. In contrast, when m is 1000, the error is large only when lambda is large, but when lambda is small the error is still relatively low.

This can be explained from the trade-off in the value lambda mentioned in lecture 6 page 11:

When lambda is small, we "pay" more for sample size, meaning high estimation error.

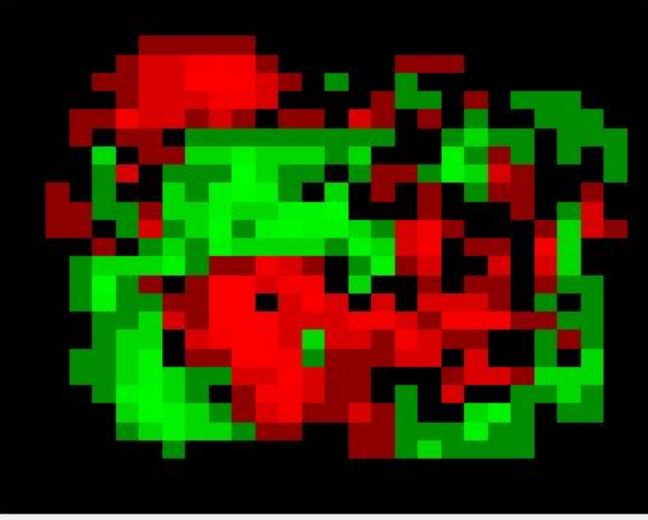
This is why when m=50, the error is high for small lambda values. When lambda is large, we pay more for the norm of w. This means that the minimization looks for w with a smaller norm, meaning large margin, which means a bigger hinge loss. This does not depend on the sample size, and this is why the error is relatively high in both cases when lambda is large.

The trend is generally similar in both lines – very low or very high lambda results in high error, whereas somewhere in the middle we achieve minimum error.

Furthermore, the height when m=1000 is lower on all points. This is expected since a bigger sample size should lead to better results.   
Another interesting finding is that the difference between the lines is very large when lambda is small. This can be explained by the trade-off mentioned in lecture 6 page 11 – when lambda is small, we have to pay more for small sample sizes.

When m is higher, the graph generally appears more smooth. TODO – explain why?

3. d.   
In the following heatmap, large positive values are represented by red color, and negative numbers with a large absolute value are represented by light green color. Numbers close to 0 are represented by black color.



3. e. The green color roughly makes a 3 shape, whereas the red roughly makes a 5 shape (laid down, like the pictures in the sample). This makes sense because 3 was labeled as -1 and 5 was labeled as 1 in our code. For each pixel, if its value in the example is bigger than w’s value, it will contribute towards a prediction as 5, and vice versa for 3.  
  
Because of this, the dominant pixels in 5 are represented in red and the dominant pixels in 3 are represented in green.

4.

