Problem 4.4.a

Page 124 of the LFD says that we normalize f so that the noise level σ^2 is automatically calibrated to the signal level. This means that no matter how we change Qf the noise will always have the same level of effect on the output.

To show the term we need to normalize by we need to rewrite the expected value.

$$\mathrm{E}_{a,x}[f^2] = \mathrm{E}_x\left[\mathrm{E}_a[f^2]\right]$$

Using the definition of variance:

$$Var(X) = \mathrm{E}[X^2] + \mathrm{E}[X]^2$$

$$Var(X) - \mathrm{E}[X]^2 = \mathrm{E}[X^2]$$

We can rewrite the first equation as:

$$\mathrm{E}_x\left[\mathrm{E}_a[f^2]\right] = \mathrm{E}_x[\mathrm{Var}_a(f) + \mathrm{E}_a[f]^2]$$

Substitute in the equation for f:

$$= \mathrm{E}_x\left[\mathrm{Var}_a\left(\sum_{q=0}^{Q} a_q L_q(x)\right) + \mathrm{E}_a\left[\sum_{q=0}^{Q} a_q L_q(x)\right]^2\right]$$

$$= \mathrm{E}_x\left[\sum_{q=0}^{Q} L_q(x)^2\,\mathrm{Var}_a(a_q) + \sum_{q=0}^{Q} a_q L_q(x)\,\mathrm{E}_a[a_q]^2\right]$$

Since, a_q is sampled from a standard normal it has var = 1 and μ =0 which gives the following:

$$= \mathrm{E}_x\left[\sum_{q=0}^{Q} L_q(x)^2\right] = \sum_{q=0}^{Q} \mathrm{E}_x\left[L_q(x)^2\right] = \sum_{q=0}^{Q} \frac{1}{2}\int_{-1}^{1} L_q(x)^2\,dx$$

Using the definition from 4.3(e):

$$\sum_{q=0}^{Q} \frac{1}{2}\int_{-1}^{1} L_q(x)^2\,dx = \sum_{q=0}^{Q} \frac{1}{2q+1}$$

Now to make $\mathrm{E}_{a,x}[(nf)^2] = 1$:

$$\mathrm{n}^2\mathrm{E}_{a,x}[f^2] = n^2 \sum_{q=0}^{Q} \frac{1}{2q+1} = 1$$

$$\boldsymbol{n = \frac{1}{\sqrt{\sum_{q=0}^{Q} \frac{1}{2q+1}}}}$$

Problem 4.4.b

To obtain g2 we need to do a nonlinear transformation on the generated data $x \in X$. This will give us $z = \phi_2(x_n) \in Z$. We use a combination of Legendre polynomials to perform the transformation. An example of this can be seen on page 129 of the LFD book.

Now we are in a higher order Z-space where we can apply linear Regression to find the optimal weight vector $\widetilde{w}_{lin}$. To get $\widetilde{w_{lin}}$ we compute the pseudo-inverse of our transformed data which gives us $\widetilde{w}_{lin} = Z^{\dagger}y$. (We use glmfit to find our w in this experiment). Finally, we can apply this separator to get:

$$g_2(x) = \tilde{g}(\phi_2(x)) = \widetilde{w}_{lin}{}^{T}\phi_2(x) = \widetilde{w}_{lin}{}^{T}z$$

We do the same thing to find g10 except we use $\phi_{10}(x_n)$ to transform our data into the Z-space.

Problem 4.4.c

To find Eout(g10) we will use the squared error measure:

$$\mathrm{E}_{out}[g_{10}(x)] = \mathrm{E}_x\left[\left(g_{10}(x) - y(x)\right)^2\right] = \mathrm{E}_x\left[\left(g_{10}(x) - f(x) + \sigma\varepsilon_n\right)^2\right]$$

Problem 4.4.d

All of the following figures were generated in matlab using colorbar and the 'jet' colormap. I then interpolated the data to smooth out the color transitions. Finally, I used caxis([-.2 .2]) for the colorbar so that we would look at the overfit measure only between -.2 and .2. I did this so that I could get something like what is found on page 124 of the LFD book.

Figure 1.

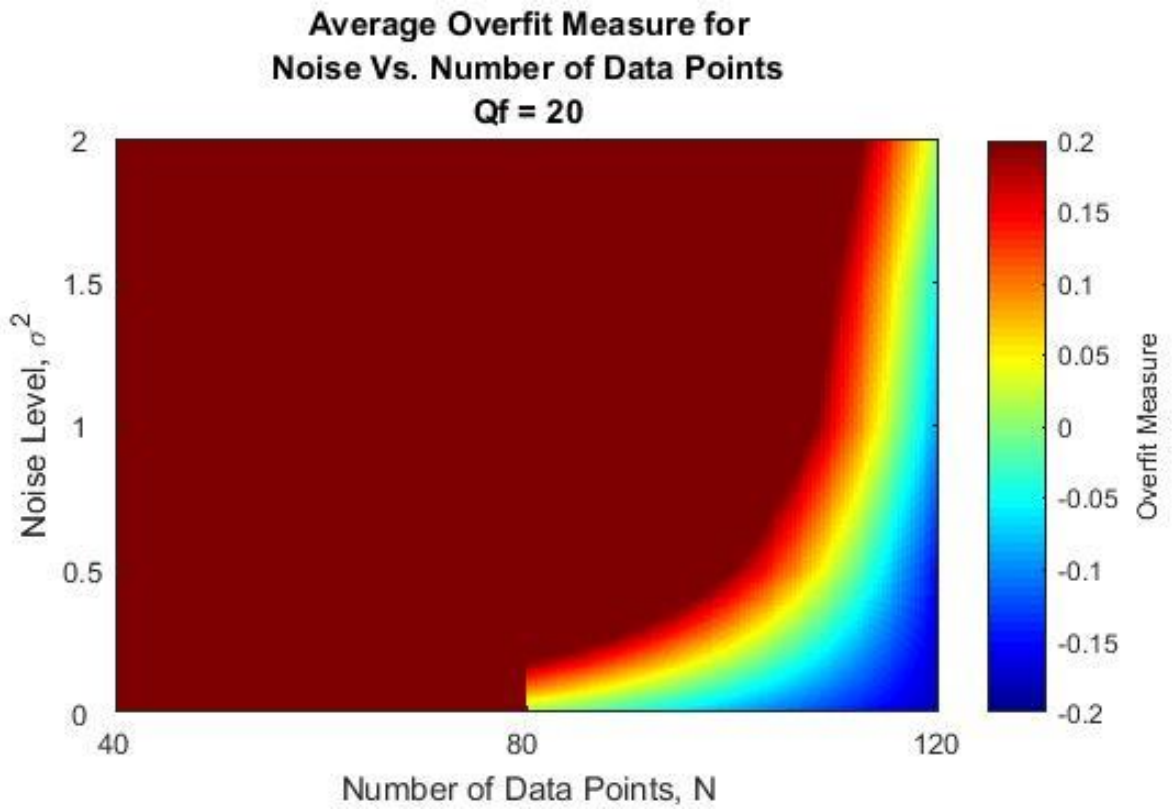

Table 1.

Average Overfit Measure, Qf= 20			
	Number of Data Points, N		
Noise Level	63.87274	0.010101	-0.17989
	24.23097	0.557619	-0.15037
	110.1931	0.864733	-0.08788
	198.5376	0.950642	-0.0496
	181.2626	1.01233	0.01756

Figure 2.

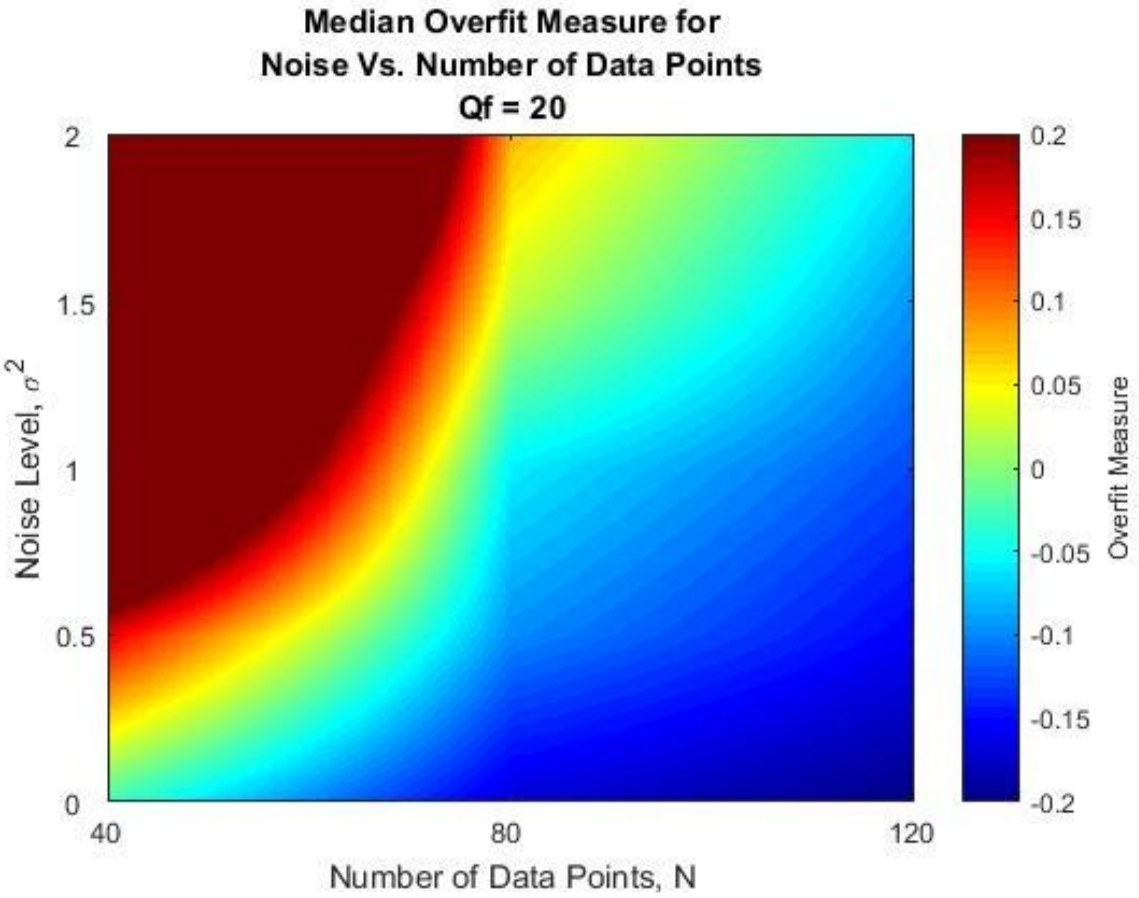

Table 2.

Median Overfit Measure, Qf= 20			
	Number of Data Points, N		
Noise Level	-0.03016	-0.1686	-0.20189
	0.155202	-0.09721	-0.15509
	0.455674	-0.05845	-0.12511
	0.649658	0.024691	-0.09373
	1.058744	0.075912	-0.05834

Figure 3.

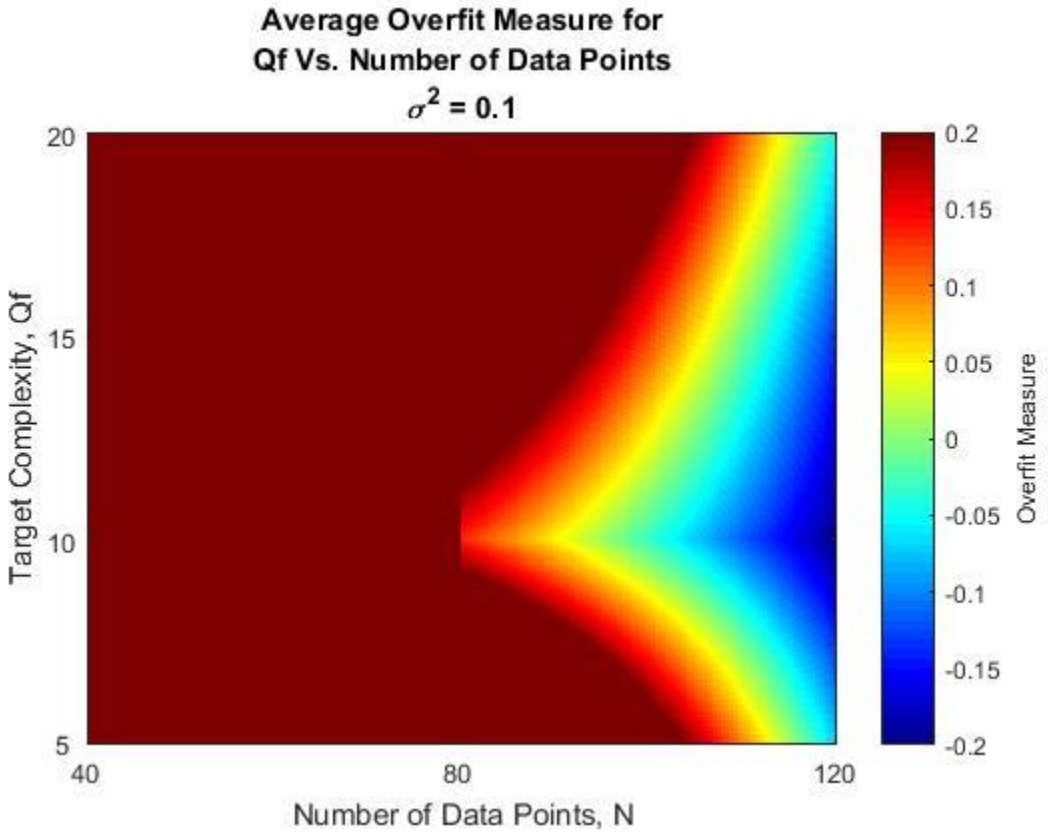

Table 3.

Average Overfit Measure, $\sigma^2 = 0.1$			
	Number of Data Points, N		
Noise Level	137.899	0.601152	-0.06843
	28.70533	0.132342	-0.19746
	76.79953	0.377466	-0.1334
	37.25359	0.592672	-0.04505
	137.899	0.601152	-0.06843

Figure 4.

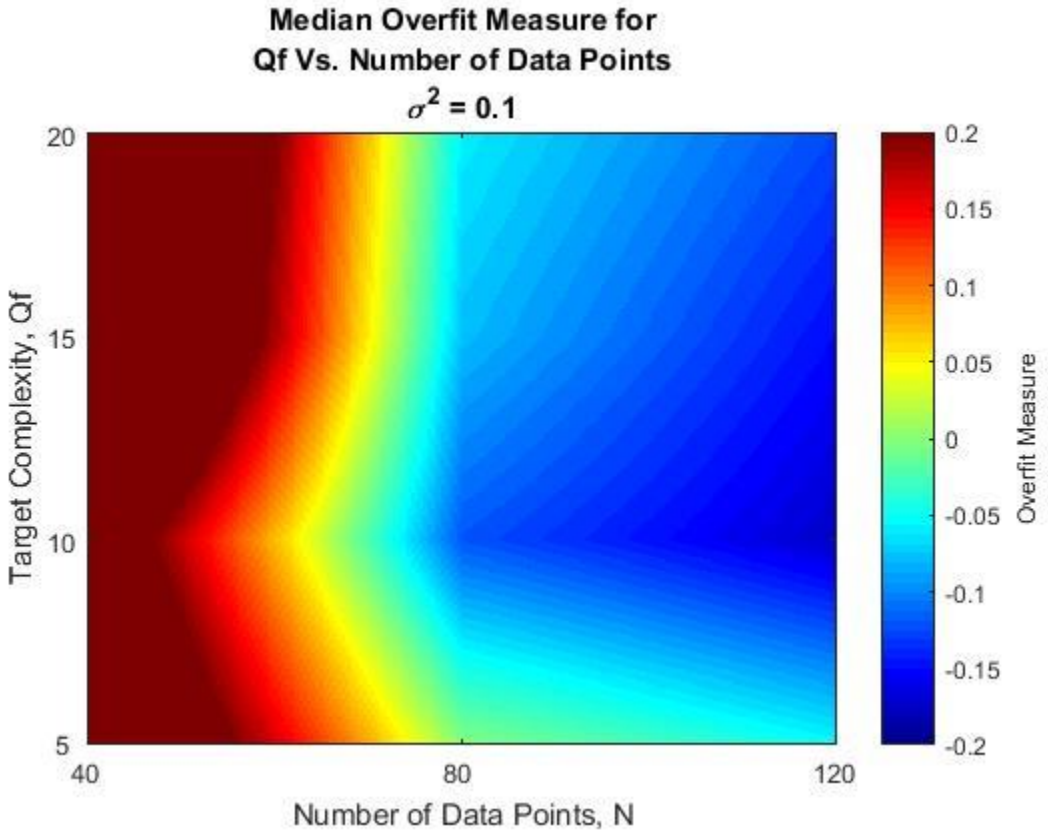

Table 4.

Median Overfit Measure, $\sigma^2 = 0.1$			
	Number of Data Points, N		
Noise Level	0.325324	-0.00088	-0.05298
	0.264669	-0.11989	-0.17398
	0.439481	-0.07829	-0.15029
	0.455674	-0.05845	-0.12511
	0.325324	-0.00088	-0.05298

There are a few insights we can draw from these figures.

Figures 1 and 2:

First, when looking at Figure 1, which uses the average, we see that even with zero noise there is still some minimum amount of data, N, needed to improve the overfit measure. If there are few data points a more complex model will still perform poorly against a complex target function with no noise.

The second is that as the number of data points increases our overfit measure generally improves for a given noise level. However, as the noise increases the number of data points we need to get a low overfit measure also increases.

As noise increases the overfit measure increases. As N increases the overfit measure decreases.

Figures 3 and 4:

From these figures we can see that, generally, as the Target Complexity Qf increases the overfit measure also increases. As N increases the over fit measure decreases. One interesting thing we can see is that as Qf approaches 10 the overfit measure improves slightly. This can be seen in the heatmap where the colors bump out to the left at 10. This is because our g10 will perform better on Qf = 10 which will lead to a smaller overfitting measure.

Mean Vs. Median

When taking the average for a low number of data points the overfit measure is often very large which can be seen in Tables 1 and 2. Through experimentation I found that this is because Eout(g10) will occasionally return very high results which throw off the total average.

When using the median, we get a much more stable set of results which are much closer to the expected results (-2 to .2). The median isn't as susceptible to the large error rates returned by instances of the experiment. So for this experiment it returns more "typical" values of the overfit measure.

For this experiment, **using the median seems more valuable for examining the impact of the Qf, σ, and N on the overfit measure.**

Problem 4.25.a

No, we shouldn't select the learner with the minimum validation error. The VC bound given for this problem on pg 163 shows that:

$$\mathrm{E}_{out}[\overline{g}_{m^*}] \leq \mathrm{E}_{val}[\overline{g}_{m^*}] + O\left(\sqrt{\frac{ln\ M}{2K}}\right)$$

This means that if one learner use a very small K and has a small validation error their Eout might still be worse than someone with a slightly larger Eval who used a very large K.

The way this new procedure is set up we don't know as much about the VC bound as the original set up.

Problem 4.25.b

Section 4.3.2 of the book tells us that using the validation set is useful in model selection. If the models are trained before seeing the validation set |Hval| = M and we can apply the VC bound for a finite hypothesis set. This allows us to choose the model with the lowest validation error to best estimate Eout.

Problem 4.25.c

First, we rearrange the Hoeffding Inequality to get:

$$P[\mathrm{E}_{out}(m^*) > \mathrm{E}_{val}(m^*) + \ \varepsilon] \leq e^{-2\varepsilon^2 K}, for\ \varepsilon > 0$$

On page 23-24 of the LFD it is shown during the proof of the Hoeffding inequality that for m=1…M:

$$P[\mathrm{E}_{out}(m^*) > \mathrm{E}_{val}(m^*) + \ \varepsilon] \leq \sum_{m=1}^{M} P[\mathrm{E}_{out}(m) > \mathrm{E}_{out}(m) + \ \varepsilon] \leq \sum_{m=1}^{M} e^{-2\varepsilon^2 K}$$

If we substitute k(ε) into the Hoeffding Bound given for multiple hypothesis:

$$Me^{-2\varepsilon^2 K} = Me^{-2\varepsilon^2\left(-\frac{1}{2\varepsilon^2}\ln\left(\frac{1}{M}\sum_{m=1}^{M} e^{-2\varepsilon^2 K}\right)\right)} = \sum_{m=1}^{M} e^{-2\varepsilon^2 K}$$

Since these two terms are equal we can write,

$$P[\mathrm{E}_{out}(m^*) > \mathrm{E}_{val}(m^*) + \ \varepsilon] \leq Me^{-2\varepsilon^2 \mathrm{k}(\varepsilon)}$$

Problem 5.4.a

i. There are a few things that we did wrong. The first problem is data snooping. We set our bound based on the past 12,500 days but we are using only the S&P 500. We are only evaluating the 500 biggest **currently traded** stocks and ignoring all others. Since they are currently traded that means they must be at least somewhat successful because they still exist on the stock market. This will lead us to a biased result and is why our bound says there is a 95% the stock will be profitable.

The second problem is that we are using 12500 days of data but some of the 50,000 stocks have come and gone within that time. Not all stocks, and probably several in the S&P 500, have not been around that long.

ii. The correct M should be 50,000 so that the bound includes all stocks that we have for the last 12,500 days. With M = 50,000 we get:

$$P[|\mathrm{E}_{in} - \mathrm{E}_{out}| > 0.02] \leq 2 * 50{,}000 * e^{-2*12{,}500*0.02^2 \approx 4.5399}$$

Within this bound tells us that there is only a %100 – %45.399 = %54.601 chance this stock will be profitable.

Problem 5.4.b

i. This is like the example for data snooping given on page 177 of the LFD book. We biased our results by only picking currently traded companies (S&P 500). We ignore companies that died off and where our buy and hold strategy probably wouldn't have performed as well. When we take this strategy to the real world we have no way of determining what companies today will last into the future and the strategy will fail. For this reason we can't conclude that it is a good strategy.

ii. We could say that the buy and hold strategy is profitable for the current S&P 500 if we could travel back in time 50 years. Otherwise, to test whether this strategy is *potentially* useful in other areas it would have to be applied to all 50,000 stocks and not just the S&P 500.