CS 541 Homework 2

Stephen Szemis

October 17, 2020

Part 1: Summary of Lectures and Thoughts

In the first lecture we started the course by setting expectations and reviewing some of the mathematical tools we would need in order to continue. I was already relatively familiar with the linear algebra notation and concepts so that was all quite easy. I definitely needed reminding when it came to some of the probability basics, so the review was helpful. And while I certainly had studied calculus in undergraduate classes, I hadn't worked with those concepts in some time. I had to sit down after some classes and review some of the calculus based proofs so that I was sure I could follow along.

Moving on to Markov's Inequality, listed bellow:

$$\text{If } X > 0,\ P(X \geq t) \leq \frac{\mathbb{E}[X]}{t}\text{for all } t > 0$$

While very helpful, it was clear from the examples in lecture that the inequality can only get a high probability argument for $X \leq 100 * t$, (or some other large value) which is not always what we want. It doesn't give very tight bounds for t.

Chebyshelv's Inequality extends Markov Inequality by using the variance, it is shown below:

$$Pr(|X - \mathbb{E}[X]| \geq t) \leq \frac{Var[X]}{t^2}$$

As can be seen, t grows exponentially, which helps to give a much tighter fit.

All of this is very reasonable and I didn't have too much trouble using these inequalities, especially when I could go back and review the lectures.

We then looked at some practical uses of Chebyshelv's Inequality, in particular in the domain of dimension reduction. The proof of random projection (as well as Johnson-Lindenstrauss Lemma) was well explained. All of this work helped immensely when it came time to implement random projection in homework 1.

Next we reviewed subspaces and null spaces. I was already pretty familiar with these concepts from other courses, so it was easy to follow.

Principal Component Analysis was our next large topic. The basic steps for PCA are as follows:

1. Run Singular Value Decomposition on our data
2. Assuming that our S diagonal elements are in non-increasing order, use the first k values of S matrix.
3. Create U prime from first k columns of U.
4. Create V prime from first k rows of V.
5. Use the new prime version of our matrixes to create new low rank representation of data.
6. Can either use $(V')^T$ or $S' * (V')^T$.
7. You can also formulate this as A * X, where X is original data and A is our U'.

So that all makes sense.

Next was applying these ideas to a recommendation system. This was a bit more confusing. We discussed why minimizing the Frobenius norm between of our actual sparse data (Z) and our filled out prediction matrix (X) makes sense. We also discussed how this is related and somewhat equivalent to the collaborative filtering approach.

There are two different types of collaborative filtering, one on for real values and one for binary values. The binary case is not simple, because we need to recover the "actual" preference values of our users, while still having a very sparse matrix. This gave us a jumping off point for discussing Maximum Likelihood Estimation.

MLE attempts to find the probability (that is recover some probability P) which gives the best likelihood of matching our observations. It sort of recovers the probability distribution of our data (though not really because we need to assume a probability distribution in order to calculate the MLE). We only looked at the binary class for MLE, since it was simpler, though MLE can be extended to a multi-class scenario.

MLE can not always be solved directly though, since it requires finding the minimum of some rather complex functions, therefore it is necessary to design an efficient algorithm for finding the minimum. Gradient Decent to the rescue! Basically, in gradient decent we "follow" the negative gradient of our function, updating our weights accordingly. It converges at a linear rate (awesome!) but can get stuck in local optimums / stationary points (not awesome).

And that's basically everything in the lectures. Obviously I've skipped writing most of the proofs / examples. In general I find the lectures very helpful, the examples are useful for showing how a theorem or concept might be applied. And the proofs have helped give a bit more formal foundation to the course. I suppose my only complaint is that many of the hand written notes the professor does in class aren't really reflected in the actual slides, so it can be difficult to locate where in which lecture a particular example was covered. Maybe just having a table of contents or something for the lectures would save me having to slowly go through each zoom recording trying to find when the professor did a certain example.

Part 2: First Homework

I'm not going to write much on this. First homework took me some time, but definitely wasn't challenging. I think it was useful for understanding the behavior of random projection and some of the inequalities we saw. Only complaint is that some of the hints the professor gave at the start of one of the lectures maybe should just be part of the homework specification.