Machine Learning 10-701 February 25, 2013
http://alex.smola.org/teaching/cmu2013-10-701/ due on March 18, 2013
Carnegie Mellon University

Homework 3

Instructions

- The homework is due in the lecture on March 18. Anything that is received after the lecture will not be considered.
- Please submit one set of notes for each of the problems and put them into a separate stack. Don't forget to add your name on each sheet.
- Alternatively, you can e-mail your solution to `10.701.homework@gmail.com`. Please write the subject as: `[Homework 3] yourandrewID` followed by the numbers of the questions you are including in the email, or "all" if you're submitting the entire homework. As before, the cutoff is the end of the lecture.
- If you are on the waitlist, please either e-mail your solution or hand it in with everyone else. However, put it into an envelope and write waitlist on it. While we cannot guarantee that you will definitely get a spot, we will give preference to students who submitted homework.
- If you submit code, it should be sufficiently well documented that the TAs can understand what is happening. Also attach pseudocode if you feel that this makes the result more comprehensible.
- The number of ◦ indicates the hardness of the problem. This is only a hint and often the hard problems simply require a bit of thinking to get the idea for the proof.

Machine Learning 10-701 February 25, 2013
http://alex.smola.org/teaching/cmu2013-10-701/ due on March 18, 2013
Carnegie Mellon University

Homework 3

1 Vapnik Chervonenkis Dimension (Ina) - 45 points

1.1 Finite function classes ○

Assume that we are given a function class $\mathcal{F}$ that contains finitely many functions, i.e. $|\mathcal{F}| < \infty$. Compute an upper bound on the VC dimension, that is, bound how many points $x_1, \ldots, x_m$ there can be at most such that for all arbitrary $y_i \in \{\pm 1\}$ there exists some $f \in \mathcal{F}$ with $\operatorname{sgn} f(x_i) = y_i$.

1.2 Gram matrix ○○

One may prove that for certain kernels such as the Gaussian RBF kernel, the kernel matrix K with $K_{ij} = k(x_i, x_j)$ has full rank, regardless of the size of the matrix K. See e.g. Michelli, Interpolation of scattered data, Constructive Approximation 1986.[1] Prove for these kernels the following function class has unbounded VC dimension:

$$\mathcal{F} := \left\{ f(x) = \sum_{i=1}^{m} \alpha_i k(x_i, x) \text{ for } x_i \in \mathcal{X} \text{ and } m \in \mathbb{N} \right\} \tag{1}$$

1.3 One parameter and infinite dimensions ○ ○ ○

Usually the number of parameters is a good quantity to guesstimate the complexity of the function class under consideration. However, this need not always be the case. Consider points $x \in \mathbb{R}$ and the class

$$\mathcal{F} := \{\sin a\pi x \text{ for } a \in \mathbb{R}\} \tag{2}$$

Show that the VC dimension of $\mathcal{F}$ is infinite. *Hint: consider placing the points x_i on $\mathbb{N}$ in special places. Can you encode the bit sequence of arbitrary labels in a suitable choice of a?* Use the above to show that for $x \in (0, 1]$ the following class also has infinite VC dimension:

$$\mathcal{F} := \{\sin a\pi/x \text{ for } a \in \mathbb{R}\} \tag{3}$$

1.4 Fat shattering and covering numbers ○ ○ ○○

In many cases such as regression or density estimation, scale insensitive quantities like the VC dimension are not ideal for measuring model complexity. In this case one can introduce the *fat shattering* VC dimension.

One definition defines $\mathrm{VC}_\epsilon[\mathcal{F}]$ to be the largest m for which there exists some $\{x_1, \ldots, x_m\} \in \mathcal{X}$ and $\{y_1, \ldots y_m\} \in \mathbb{R}$ such that for all $z_i \in \{\pm 1\}$ we can find an $f \in \mathcal{F}$ satisfying

$$z_i(f(x_i) - y_i) \geq \epsilon. \tag{4}$$

In other words, for $z_i = -1$ we have $f(x_i) \leq y_i - \epsilon$ and for $z_i = 1$ we have $f(x_i) \geq y_i + \epsilon$ for any arbitrary choice of $\{z_1, \ldots, z_m\}$. The covering number $N_\epsilon(\mathcal{F})$ of a function class $\mathcal{F}$ is defined as the number of ϵ-balls that are needed to approximate $\mathcal{F}$ to ϵ precision.

- Compute a boud on $\mathrm{VC}_\epsilon(\mathcal{F})$ as a function of $N_\epsilon(\mathcal{F})$ provided that the ϵ-approximation holds uniformly on $\mathcal{X}$.
- Assume that $\mathcal{F} = \{f | f = \langle w, x \rangle \text{ and } \|x\|_2 \leq 1\}$ and that $\|x\|_2 \leq 1$. Compute a bound on $N_\epsilon(\mathcal{F})$ if we are only interested in an ϵ-cover on the m points $x_1, \ldots, x_m$. A crude bound is sufficient. *Hint: it suffices to cover the ℓ_2 ball in $\mathbb{R}^m$ as spanned by the points x_i.*

[1] http://www.maths.gla.ac.uk/~tl/Micchelli_Interpolation.pdf

Homework 3

2 40 Shades of Blue (Junier) - 30 points

Google wants to try out a new shade of blue.[2] To ensure that the most amazing blue is chosen, it tries them out on users to see whether they're more likely to click on a given button. Obviously, it wants to complete this experiment as quickly as possible to provide the bluest blue to everyone. Your task is to determine how many users need to see this before the designer in chief can pick a color and to avoid insignificant results.[3]

The experiment works as follows: a user is given a page containing a particular shade of blue and Google records clicks on a user interface element with this hue. These things are then logged and the number of clicks is counted. In other words, each hue $i \in \{1, \dots 40\}$ is displayed to m random visitors and we observe m_i clicks for each i as a result. We want to find the i corresponding to the largest click probability.

2.1 Confidence Bounds ○○

You may assume that the click probability never exceeds 0.1. Using this fact, how many sessions do you need to establish the click probability for all 40 shades with error 0.001? Derive bounds using the following:

1. Markov inequality
2. Chebyshev inequality
3. Hoeffding's inequality
4. Bernstein's inequality, that is, using the inequality

$$\Pr\left\{\sum_{i=1}^{m} x_i > \epsilon\right\} \leq \exp\left(-\frac{\epsilon^2}{2\sum_i \mathbf{E}[x_i^2] + 2M\epsilon/3}\right) \tag{5}$$

Here x_i are indepdentent zero-mean random variables with $|x_i| \leq M$. *Hint: to obtain zero-mean random variables use the transformation $x_i \leftarrow x_i - \mathbf{E}[x_i]$. Moreover, the upper bound on the click probability allows us to compute a bound on the variance.*

2.2 Dependent Outcomes ○ ○ ○

Now assume that we are giving the users a choice between 40 different outcomes, e.g. by asking them directly which color they prefer. In other words, in each session we observe one of 40 events.

1. How many sessions do we need to establish with accuracy 0.001 the popularity of *all* colors? Use Hoeffding's or Bernstein's inequality to derive this.
2. Argue how we could get away using fewer sessions than the above number if we only wanted to find the *most popular* color? No need to derive a detailed strategy, just explain what property you could exploit for estimation.

[2] http://iterativepath.wordpress.com/2012/10/29/testing-40-shades-of-blue-ab-testing/.
[3] http://imgs.xkcd.com/comics/significant.png

3 Sketches (Mu) - 40 Points

A hash function[4] $h : \mathcal{X} \to \{1, \dots n\}$ is a mapping that converts items from some domain $\mathcal{X}$ to the range of integers between 1 and n. For our purposes we consider this function as a parametric random variable, just with the added benefit that we can look up its value at any time (again). That is, the probability that $x_1, \dots, x_j$ maps into any elements from $\{1, \dots n\}$ is uniform.

We now use this to design an algorithm for sketching the frequency of items in a data stream. That is, we want to have an approximate estimate of how frequently we see a particular item. For this we need insert and query operations.

Init(k,n)
Create array $M \in \mathbb{R}^{k \times n}$ with $M[i, j] = 0$ for all i, j.

Insert(x)
for $i = 1$ **to** k **do**
  $M[i, h(i, x)] \leftarrow M[i, h(i, x)] + 1$
end for

Query(x)
$r = \infty$
for $i = 1$ **to** k **do**
  $r \leftarrow \min(r, M[i, h(i, x)])$
end for
return r

3.1 Lower Bound ∘

Denote by m_x the number of times we actually observed x.

1. Prove that $M[i, h(i, x)] \geq m_x$.
2. Prove that **Query(x)** returns an upper bound on n_x.

3.2 Expectation ∘∘

Denote by m the total number of observations that we made. Compute the expected value of $M[i, h(i, x)]$ as a function of m_x, m and n. That is, take the expectation over all hash functions by treating the probability that $h(i, x) = h(i, x')$ for $x \neq x'$ as a random variable with probability n^{-1}.

3.3 Upper bound $M[i, h(i, x)]$ ∘∘

Use Markov's inequality to bound the probability that $M[i, h(i, x)] > n_x + em/n$.

3.4 Upper bound query(x) ∘ ∘ ∘

Now assume that we want to bound the probability that **query(x)** considerably overestimates n_x. For this purpose derive a bound on the probability that **query(x)** $> n_x + em/n$. *Hint: Use the previous result and exploit the fact that conditioned on the data distribution the $M[i, h(i, x)]$ are independent random variables for different i.*

Note: This is one of the most effective sketching algorithms for counts of observations. You can check that the sketch is tolerant to removals as long as the final number of counts is nonnegative.

[4] http://en.wikipedia.org/wiki/Hash_function

4 Experiments with Random Variables (Xuezhi) - 35 Points

Provide well-commented code for your problems.

4.1 Central Limit Theorem ○○

Empirically verify that the central limit theorem holds for a number of random variables. Assume that

$$x_{ij} \sim p(x) \text{ and } z_i := \frac{1}{\sqrt{n}} \sum_{j=1}^{n} x_{ij}. \tag{6}$$

Now, for $n \in \{1, 3, 10, 30, 100, 300, 1000, 3000\}$ generate 10000 random variables z_i and plot a histogram using 500 bins.

1. Do this for the Normal distribution $\mathcal{N}(0, 1)$ with mean 0 and variance 1.
2. Do this for the uniform distribution $U[-1, 1]$ over $[-1, 1]$.
3. Do this for the centered Poisson distribution, that is $x = v - 1$ where $v \sim \text{Poi}(1)$.

Provide plots of the histograms. It is OK if you use libraries to draw from Poisson and Gaussians.

4.2 Tails from the Central Limit Theorem ○

Plot the 0.05 and 0.95 quantiles of z_i for the Uniform distribution as a function of n.

4.3 Sampling with replacement ○ ○ ○

Assume that we have n items. We draw with replacement αn from this set.

1. Derive an the approximation $e^{-\alpha}$ for the fraction of items not drawn even once. *Hint: use* $(1-x) < e^{-x}$ *for* $x > 0$ *and the limit definition of* e^{-1}.
2. Empirically verify your bound for $n \in \{1, 3, 10, 30, 100, 300, 1000, 3000\}$ and for $\alpha \in \{0.5, 1, 2, 5\}$.