1 Notes related to probability concepts

1.1 Sketch

The techinique is to apply a ”linear projection on the fly” that converts high dimensional data into smaller dimensional data and then extract a quantity of interest [4] in the lower dimension.
Can also call the smaller dimensional data as summary.

1.1.1 Formal definition

$V := (x_0, x_1, ...x_n)$
$h :=$ any hash function
$x :=$ any vector

$V^n \rightarrow V^m$
$x \rightarrow h(x_i)$

```
Sketch :: [a] -> (a -> a) -> [a]
Sketch x h = foldl h x
```

obs: h deals per type of x not the whole vector
obs2: we use foldl and not map because a reduction in the size of the vector happens and map deal with same size vectors. [6]

1.2 Estimation

Before we can discuss estimation we must explain what is the process of statical inference. well inference is guessing something based on previous information note the word **guessing** there's no guarantee the right answer will be reached but an ”accurate approximation” nonetheless.

Thus estimation is a technique to achieve statical inference, the idea is that estimation allow us to guess the value of parameters that model some probability distribution

What to know about the quality of estimations:

1.2.1 μ as symbol for confidence interval

the idea here is that the estimation can fall in a range, generally between 95%-99% that guarantees a confidence level

1.2.2 n as symbol for sample size

Well for each estimation we assume a minimum ammount of points in the data also called sampling that will allow the estimation process to happen and fall in the confidence interval

[2]

1.3 Estimator

A fancy name for someting very simple an function that receives a set of samples and output a estimate that is a parameter to model the distribution here some fancy names for categorizing these functions

- Biased: underestimation or overestimation
- Efficient: small variance
- Invariant: hard to change even after transformations, imagine apply a increase in each sampled datapoint the estimation would stay the same
- Shrinkage: raw estimation
- Sufficient: enough samples to extract a confidence level compliant estimation
- unbiased: not over/under estimation

OBS: statistic is not the same as estimation, estimation is a way to acquire an statistic, there are other ways as well [3]

1.4 (ϵ, δ)-approximation

Another notational way to say that estimation error is related to ϵ and the confidence interval of that information is related to the δ parameter
Thus one could say with $\epsilon < 0.2$ we guarantee with $\delta > 0.9$ that such estimate is accurate.

map absolute value of error to ϵ
map confidence interval to δ

1.4.1 Formal definition

$$\mathbb{P}[|\phi - t| > t] \leq \delta \tag{1}$$

Notes based on the following resources:
[1]

2 The Morris Counter

The Morris counter is an algorithm created to allow approximate calculations of values in devices that can't represent such numbers due to physical limitations, amount of bits available.

2.1 What is approximate counting?

The concept of approximate counting is deducing the total number of events that occurred but storing as minimal events as possible to allow such deduction. To compute the error parameters we mostly look at the standard deviation, that give us a measure of accuracy/trustworthiness.

Effectively we can double, triple , n-increase the tracked value but storing a fraction of it size, indeed a very powerful technique.

2.2 So why the Morris counter?

Might not be obvious but a considerable difference in the error according to the sample size implies less guarantees, thus a method that guarantees for the average case a stable margin of error, that is won't oscillate is a good value proposition.

we use the following equation as an starting point to understand the algo

$$v_n = ln(1 + n) \tag{2}$$

the choice for logarithm is due to the fact that v increases much faster than n, thus low amount of events tracked guarantee better approximations and average case error is constant.

However the value of the amount tracked,v, is not an integer and must be rounded up/down.

2.3 How to determine the rounding

2.3.1 Cost innacuracy

Assume an increase in **r**, either the value of v will be undershoot or overshoot, we need an heuristic to deal with this, thus the concept of a marker, **d** to help with this we also use a rng approach to generate the parameter **r** and if r is larger or smaller than the mark d, then do the following:

$$d = \frac{1}{n_{v+1} - n_v} \tag{3}$$

if larger: increase the counter
if smaller: keep current value

proof of this is here: [5]

then we get to this final equation

$$v = \frac{log(n+1)}{log(2)} \tag{4}$$

2.4 The main advantage of this approach

Saving space, the amount of bits required to be tracked is $log(n)$ and bits implies binary counting that is with n bits we can track 2^n states that eventually map to numbers, thus a binary morris counter actually is $O(log(log(n)))$ in spatial complexity, because is not dealing with integers, but bits.

References

[1] A. Bhayani. Morri's algorithm for approximate counting. Blog post.

[2] B. Gerstman. 5: Introduction to estimation. Notes link .

[3] HowTo. Estimator: Simple definitions and examples. BlogPost link .

[4] A. McGregor. Crash course on data stream algorithms, part i: Basic definitons and numerical streams. Slide link.

[5] R. morris. Counting large numbers of events in small registers. *Programming Techniques*, Link to paper.

[6] A. Potdar and H. K. Borah. Streaming algorithms. Slide link .