COMP4121 Lecture Notes - Topic 4

More fixed points algorithms: Iterative Filtering algorithms for robust aggregation of sensor data

LiC: Aleks Ignjatovic

ignjat@cse.unsw.edu.au

THE UNIVERSITY OF
NEW SOUTH WALES

School of Computer Science and Engineering
The University of New South Wales
Sydney 2052, Australia

You might want to review basic probability theory from the lecture notes available on the course website.

1 A Review of Basic Statistical Concepts

- **A random vector** is a sequence of n random variables $(X_1,\dots,X_n)$; we denote the (multivariate) PDF of such a vector by $f_{\mathbf{X}}(x_1,\dots,x_n)$.

- Random variables $X_1,\dots,X_n$ are **independent** if for all subsets $A_1,\dots,A_i \subset \mathbb{R}$,

$$\mathcal{P}(X_1 \in A_1,\dots,X_n \in A_n) = \prod_{i=1}^{n} \mathcal{P}(X_i \in A_i); \tag{1.1}$$

this happens just in case

$$f_{\mathbf{X}}(x_1,\dots,x_n) = \prod_{i=1}^{n} f_{X_i}(x_i). \tag{1.2}$$

- If random variables $X_1,\dots,X_n$ are independent and have the same probability distribution $F(x)$, we say that $X_1,\dots,X_n$ are **IID (independent identically distributed)** random variables, or **random sample of size n from F**.

- The **sample mean** is the random variable defined by $\overline{X} = \dfrac{X_1+\ldots+X_n}{n}$.

Assume $X_1,\dots,X_n$ are IID with mean $E(X_i)=\mu$ and variance $V(X_i)=\sigma^2$; then, since X_i are equally distributed, the expected value of $\overline{X}$ and the variance $V(\overline{X})$ satisfy

$$E(\overline{X}) = \frac{1}{n}\sum_{i=1}^{n} E(X_i) = \frac{1}{n}\, n\mu = \mu \tag{1.3}$$

and using this we get

$$\begin{aligned} V(\overline{X}) =& E(\overline{X}-E(\overline{X}))^2 = E(\overline{X}-\mu)^2 = E\left(\frac{\sum_{i=1}^{n}(X_i-\mu)}{n}\right)^2 = \\ & E\left(\frac{1}{n^2}\sum_{i=1}^{n}(X_i-\mu)^2 + \frac{2}{n^2}\sum_{i\neq j}^{n}(X_i-\mu)(X_j-\mu)\right) = \\ & \frac{1}{n^2}\left(\sum_{i=1}^{n}E(X_i-\mu)^2 + 2\sum_{i\neq j}^{n}E((X_i-\mu)(X_j-\mu))\right); \end{aligned}$$

We now use the fact that if X_i, X_j are independent then

$$E((X_i-\mu)(X_j-\mu)) = E(X_i-\mu)E(X_j-\mu) = 0\times 0 = 0$$

Thus

$$V(\overline{X}) = \frac{1}{n^2}\sum_{i=1}^{n}E(X_i-\mu)^2 = \frac{n\sigma^2}{n^2} = \frac{\sigma^2}{n}. \tag{1.4}$$

Note that this explains why taking the mean of multiple measurements provides more accurate value than a single measurement: if the variance of a single measurement is σ^2 then the varianve of the mean of n measurements is only σ^2/n.

Assume now that we have n measurements X_i of a quantity using the same instrument and want to estimate the variance of the instrument. Since the expected value of $\overline{X}$ is μ one might think that

$$E\left(\frac{1}{n}\sum_{i=1}^{n}(X_i-\overline{X})^2\right) \overset{??}{=} E\left(\frac{1}{n}\sum_{i=1}^{n}(X_i-\mu)^2\right) = \frac{1}{n}\sum_{i=1}^{n} E(X_i-\mu)^2 = \frac{n\,\sigma^2}{n} = \sigma^2$$

which would make $\frac{1}{n}\sum_{i=1}^{n}(X_i-\overline{X})^2$ an unbiased estimator for the variance of each of X_i, but this is not quite so - the "equality" with the question marks fails, because we do not take into account that in fact $\overline{X}$ is not quite equal to μ, making the expected value of the lefthand side slightly smaller than σ^2 because $\eta = \overline{X}$ in fact minimises the value of $s(\eta) = \sum_{i=1}^{n}(X_i - \eta)^2$. An unbiased estimate of the **sample variance** is given by

$$S_n^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \overline{X})^2. \tag{1.5}$$

To see that the expected value of the sample variance is equal to the variance σ^2 of all X_i, note that

$$E\left(\sum_{i=1}^{n}(X_i - \overline{X})^2\right) = E\left(\sum_{i=1}^{n}(X_i^2 - 2X_i\overline{X} + \overline{X}^2)\right) = \tag{1.6}$$

$$E\left(\sum_{i=1}^{n} X_i^2 - 2\sum_{i=1}^{n} X_i\overline{X} + \sum_{i=1}^{n}\overline{X}^2\right) = E\left(\sum_{i=1}^{n} X_i^2 - 2n\overline{X}^2 + n\overline{X}^2\right) =$$

$$E\left(\sum_{i=1}^{n} X_i^2 - n\overline{X}^2\right) = \sum_{i=1}^{n} E(X_i^2) - n\,E(\overline{X}^2) = nE(X_1^2) - n\,E(\overline{X}^2) \tag{1.7}$$

We now use the fact that for every random variable Y with mean μ and standard deviation σ we have

$$\begin{aligned}\sigma^2 =& E(Y-\mu)^2 = E(Y^2 - 2\mu Y + \mu^2) = E(Y^2) - 2\mu E(Y) + \mu^2 = \\ & E(Y^2) - 2\mu^2 + \mu^2 = E(Y^2) - \mu^2 \end{aligned}\tag{1.8}$$

i.e., $E(Y^2) = \sigma^2 + \mu^2$. Thus, continuing (1.7) and using (1.3) and (1.4)

$$\begin{aligned}E\left(\sum_{i=1}^{n}(X_i - \overline{X})^2\right) &= n(\sigma^2 + \mu^2) - n\,E(\overline{X}^2) \\ &= n(\sigma^2 + \mu^2) - n\left(E(\overline{X} - \mu)^2 + \mu^2\right) = \\ &= n(\sigma^2 + \mu^2) - n\left(\frac{\sigma^2}{n} + \mu^2\right) = \\ &= (n-1)\sigma^2 \end{aligned}\tag{1.9}$$

which clearly implies that $E(S_n^2) = E\left(\frac{1}{n-1}\sum_{i=1}^{n}(X_i - \overline{X})^2\right) = \sigma^2$, i.e., that S_n^2 is an unbiased estimator for the variance of X.

2 Maximum Likelihood Estimation

The likelihood of a set of parameter values, θ, given outcomes $\mathbf{x}$, is equal to the probability of those observed outcomes given those parameter values. The likelihood function is defined differently for discrete and continuous probability distributions.

- **Likelihood in case of a discrete probability distribution.** Let X be a random variable with a discrete probability distribution p depending on a parameter θ. Then the function

  $$\mathcal{L}(\theta|x) = p_\theta(x) = P_\theta(X = x),$$

  considered as a function of θ, is called the likelihood function of θ, given the outcome x of the random variable X.

- **Likelihood in case of a continuous probability distribution.** Let X be a random variable with a continuous probability distribution with density function f depending on a parameter θ. Then the function

  $$\mathcal{L}(\theta|x) = f_\theta(x),$$

  considered as a function of θ, is called the likelihood function of θ, given the outcome x of X. In case of several random variables, the likelihood function is equal to the joint probability density of the vector $\mathbf{X} = (X_1, \cdots, X_n)$, but seen as a function of unknown parameters of the distribution of X_i with the values of X_i treated as constants. Thus, if X_i are IID, then the likelihood is just the product of the values of the density function at the corresponding values X_i:

  $$\mathcal{L}_n(\theta|\mathbf{X}) = \prod_{i=1}^{n} f_\theta(X_i)$$

To make dependence on the unknown parameters explicit we instead write

$$\mathcal{L}_n(\theta|\mathbf{X}) = \prod_{i=1}^{n} f(X_i;\theta)$$

One way to choose the values of the parameters is to choose the values which *maximise the likelihood function*, with the intuition behind that we choose the values of the parameters for which the outcome we have is the most likely to happen. Note that likelihood is NOT the same as probability; in fact it has a somewhat "reverse" role in the following sense:

- if we have a coin with the probability of getting a head equal to p, than we can calculate the *probability* to get a particular outcome if we toss it 16 times, say the probability that we will get $HHTHTHHHTTHHTHHT$.
- if we have already performed the experiment with a coin we know nothing about and observed such an outcome, we can now ask the "reverse question": for what p is the probability of the observed outcome maximal, i.e., for what value of p is such outcome *most likely*?

Note that if we have gotten 10 heads it is intuitively clear that such outcome is most likely to have come from a coin for which the probability p of getting a head is $\frac{10}{16}$. To verify such a hypothesis, we compute the probability of such an outcome, i.e., to get 10 heads in that particular order, as a function of p:

$$\mathbb{P}(p) = p^{10}(1-p)^6.$$

To find when such probability is the largest, we look for the stationary points of $\mathbb{P}(p)$, i.e., for p such that $\frac{\partial \mathbb{P}}{\partial p} = 0$. Since

$$\begin{aligned}\frac{\partial \mathbb{P}}{\partial p} &= 10p^9(1-p)^6 + p^{10}\cdot 6(1-p)^5(-1)\\ &= p^9(1-p)^5(10(1-p)-6p)\\ &= p^9(1-p)^5(10-16p),\end{aligned}$$

$\mathbb{P}(p)$ has a maximum value for $16p = 10$, i.e., for $p = \frac{10}{16}$.

The Maximum Likelihood Estimator (MLE) is the estimator which chooses the values of the parameters involved which make the observed outcome most likely. More precisely, for discrete random variables the MLE indeed chooses for the parameters those values which maximize the probability of the observed outcome; for continuous random variables the MLE chooses the parameters which maximise the *probability density.* Note that, while $\mathcal{L}_n(\theta|\mathbf{X})$ makes the probability of the outcome maximal, is not necessarily a probability distribution on the space of the possible values of the parameters.

A ML estimate is usually good for large number of samples, but it can perform really badly if we have only a small number of samples. As an example, consider the following problem.

Assume that I have given you a box which contains n balls which are numbered consecutively 1 to n, but I do not tell you what n is, i.e., how many balls there are inside. You are allowed to draw one single ball and look at its number, and then you have to estimate how many balls there are inside.

Assume that you drew the ball numbered k. Since all balls are equally likely, if there are n balls inside, then the probability of drawing any particular ball is $1/n$. Thus, the event that you drew ball k has the highest possible probability if $1/n$ is as large as possible, i.e., when n is as small as possible. Since you know that there are at least k balls inside, the MLE estimate for the number of balls in the box is $n = k$, i.e., the MLE estimator in this case is $N(X) = X$, or simply the value of the sample.

What is the mean of this estimator? The expected value of X, i.e., $\mu = E(X)$ is then given by

$$\mu = \sum_{i=1}^{n}\left(i \times \frac{1}{n}\right) = \frac{n(n+1)}{2n} = \frac{n+1}{2}.$$

Thus, in this case the MLE estimator is extremely biased, because its expected value is only about one half of the true value n of the number of balls inside the box! If you instead use the estimator $Y(X) = 2X-1$, then the expected value of Y is

$$\sum_{i=1}^{n} \frac{2i-1}{n} = \frac{2\sum_{i=1}^{n} i}{n} - \frac{\sum_{i=1}^{n} 1}{n} = \frac{2n(n+1)}{2n} - 1 = n$$

and so this estimator is unbiased – much better than the MLE. This does not happen for large samples; it can be shown that as the size of the sample increases, ML estimate approaches the best possible estimate.

Example: Sensor data aggregation

Assume that we wish to estimate the value of some quantity θ which is measured by n sensors; let the readings of these sensors be represented by random variables $X_1, \ldots, X_n$. We will assume that these sensors do not have any systematic errors, i.e., that they do not have a *bias.* This means that the expected value $E(X_i) = \mu$ of each sensor is equal to the true value of the measured quantity.

- **Case 1:** Assume that all sensors have a normal distribution with the same standard deviation σ, i.e.,[1] $X_i \sim \mathcal{N}(\mu, \sigma)$. We take all the measurements $X_1, \ldots, X_n$ and would like to estimate the true value of the quantity measured and the value of σ. Since sensors are unbiased, this means that we want to estimate μ and σ. Let us also assume that sensor errors are independent from each other; in this case

$$\begin{aligned}\mathcal{L}_n(\mu, \sigma|\mathbf{X}) =& \prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}}\, \mathrm{e}^{-\frac{1}{2}\,\frac{(X_i-\mu)^2}{\sigma^2}} = \frac{1}{(2\pi\sigma^2)^{n/2}} \prod_{i=1}^{n} \mathrm{e}^{-\frac{1}{2}\,\frac{(X_i-\mu)^2}{\sigma^2}} \\ =&(2\pi)^{-n/2}\sigma^{-n}\, \mathrm{e}^{-\frac{1}{2}\sum_{i=1}^{n}\frac{(X_i-\mu)^2}{\sigma^2}} \qquad (2.1)\end{aligned}$$

Let us now compute the MLE estimator in this case.

Taking X_i as fixed constants (the observed outcomes of an experiment) in order for the likelihood function of the outcome to have an extremal point the partial derivatives with respect to μ and σ must be equal to zero. Thus,

$$\frac{d}{d\mu}\mathcal{L}_n(\mu, \sigma|\mathbf{X}) =(2\pi\sigma^2)^{-n/2}\, \mathrm{e}^{-\frac{1}{2}\sum_{i=1}^{n}\frac{(X_i-\mu)^2}{\sigma^2}} \left(-\frac{1}{2}\left(2(-1)\sum_{i=1}^{n}\frac{(X_i-\mu)}{\sigma^2}\right)\right) = 0; \qquad (2.2)$$

so $\frac{d}{d\mu}\mathcal{L}_n(\mu, \sigma|\mathbf{X}) = 0$ just in case $\sum_{i=1}^{n}\frac{(X_i-\mu)}{\sigma^2} = 0$, i.e.,

$$\sum_{i=1}^{n}(X_i-\mu) = \sum_{i=1}^{n} X_i - \sum_{i=1}^{n}\mu = \sum_{i=1}^{n} X_i - n\mu = 0$$

Thus, we get $\mu = \frac{\sum_{i=1}^{n} X_i}{n}$, i.e., the MLE for the expected value μ of X is just the mean. Recall that we have shown that the mean is an **unbiased estimator** of the expected value of the random variables X_i, see (1.3). Similarly, we get

$$\begin{aligned}\frac{d}{d\sigma}\mathcal{L}_n(\mu, \sigma|\mathbf{X}) =& -n(2\pi)^{-n/2}\sigma^{-n-1}\, \mathrm{e}^{-\frac{1}{2}\sum_{i=1}^{n}\frac{(X_i-\mu)^2}{\sigma^2}} + \qquad (2.3)\\ & (2\pi)^{-n/2}\sigma^{-n}\, \mathrm{e}^{-\frac{1}{2}\sum_{i=1}^{n}\frac{(X_i-\mu)^2}{\sigma^2}} \left(-\frac{1}{2}\sum_{i=1}^{n}(X_i-\mu)^2\right)(-2\sigma^{-3}) \qquad (2.4)\end{aligned}$$

After some cancelations, we see that $\frac{d}{d\sigma}\mathcal{L}_n(\mu, \sigma|\mathbf{X}) = 0$ just in case

$$-n + \sigma\sum_{i=1}^{n}(X_i-\mu)^2\sigma^{-3} = 0 \quad \Leftrightarrow \quad \sigma^2 = \frac{\sum_{i=1}^{n}(X_i-\mu)^2}{n}. \qquad (2.5)$$

Thus, Maximum Likelihood estimates for the mean and the variance of X are given by

$$\begin{aligned}\mu = \overline{X} &= \frac{X_1 + \ldots + X_n}{n};\\ \sigma^2 &= \frac{\sum_{i=1}^{n}(X_i-\mu)^2}{n}.\end{aligned}$$

[1] $X \sim F$ and $X \sim f$ denote that X has probability distribution $F(x)$ (probability density $f(x)$, respectively); if X, Y are both random variables, then $X \sim Y$ means that they share the same probability distribution.

Substituting μ from the first equation into the second equation we get

$$\sigma^2 = \frac{\sum_{i=1}^n (X_i - \mu)^2}{n} = \frac{\sum_{i=1}^n (X_i - \overline{X})^2}{n}.$$

Thus, the MLE for the variance σ^2 given by (2.5) is not even unbiased, because, as equation (1.9) shows,

$$E\left(\frac{\sum_{i=1}^n (X_i - \overline{X})^2}{n}\right) = \frac{n-1}{n} E\left(\frac{\sum_{i=1}^n (X_i - \overline{X})^2}{n-1}\right) = \frac{n-1}{n}\,\sigma^2,$$

i.e., MLE estimator for the variance has a negative bias equal to $-\frac{\sigma^2}{n}$.

As we have mentioned, in general, MLE estimators might have a bias, but as the number of samples increases, their "accuracy" approaches the optimal one, and their "dispersion" around the expected value approaches a normal distribution.[‡]

• **Case 2:** Assume that the errors of readings $X_1, \ldots, X_n$ of our n sensors are independent and unbiased, and all have normal distributions but with different and **known** standard deviations $\sigma_1, \ldots, \sigma_n$ (for example, we have tested them all in a lab doing many measurements and comparing their readings with the true value obtained from a precise instrument). Assume that, using these sensors, we have obtained measurements $X_1, \ldots, X_n$, which we would like to use to estimate the true value of the quantity being measured.

Since the standard deviations of sensors are now known, in this case the likelihood function has only one parameter, the expected value μ:

$$\mathcal{L}_n(\mu|\mathbf{X}) = \prod_{i=1}^n \frac{1}{\sigma_i\sqrt{2\pi}}\ \mathrm{e}^{-\frac{1}{2}\frac{(X_i-\mu)^2}{\sigma_i^2}} = \left(\prod_{i=1}^n \frac{1}{\sigma_i\sqrt{2\pi}}\right)\ \mathrm{e}^{-\frac{1}{2}\sum_{i=1}^n \frac{(X_i-\mu)^2}{\sigma_i^2}} \tag{2.6}$$

Differentiating with respect to μ and setting the derivative equal to zero we get

$$\frac{d}{d\mu}\mathcal{L}_n(\mu|\mathbf{X}) = \left(\prod_{i=1}^n \frac{1}{\sigma_i\sqrt{2\pi}}\right)\ \mathrm{e}^{-\frac{1}{2}\sum_{i=1}^n \frac{(X_i-\mu)^2}{\sigma_i^2}}\ \sum_{i=1}^n \frac{(X_i-\mu)}{\sigma_i^2}$$

$$\frac{d}{d\mu}\mathcal{L}_n(\mu|\mathbf{X}) = 0 \quad \Leftrightarrow \quad \sum_{i=1}^n \frac{X_i}{\sigma_i^2} - \mu\sum_{i=1}^n \frac{1}{\sigma_i^2} = 0 \quad \Leftrightarrow \quad \mu = \frac{\sum_{i=1}^n \frac{X_i}{\sigma_i^2}}{\sum_{i=1}^n \frac{1}{\sigma_i^2}}$$

i.e.,

$$\mu = \sum_{i=1}^n \frac{\frac{1}{\sigma_i^2}}{\sum_{j=1}^n \frac{1}{\sigma_j^2}}\,X_i \tag{2.7}$$

We see that in this case the MLE is a weighted average of readings of all sensors, with weights inversely proportional to their variances. Note that the sum in the denominator makes all weights add up to 1, so that we get a correct weighted average, with values between the smallest and the largest of all X_i. Note also that such estimator does not throw away reading of any sensor, no matter how large its variance might be, but uses all the readings giving lower weights to the readings of the less accurate sensors.

Also, the expected value of our estimator

$$M(\mathbf{X}) = \sum_{i=1}^n \frac{\frac{1}{\sigma_i^2}}{\sum_{j=1}^n \frac{1}{\sigma_j^2}}\,X_i$$

i.e., $E(M(\mathbf{X}))$ is equal to

$$E(M(\mathbf{X})) = \sum_{i=1}^n \frac{\frac{1}{\sigma_i^2}}{\sum_{j=1}^n \frac{1}{\sigma_j^2}}\,E(X_i) = \sum_{i=1}^n \frac{\frac{1}{\sigma_i^2}}{\sum_{j=1}^n \frac{1}{\sigma_j^2}}\,\mu = \mu$$

[‡] Defining what such "accuracy" and "dispersion approaching normal" mean requires more mathematical machinery then what we can afford here.

and consequently the estimator is unbiased. Let us now compute its variance:

$$V(M(\mathbf{X})) = E\left(\sum_{i=1}^{n} \frac{\frac{1}{\sigma_i^2}}{\sum_{j=1}^{n}\frac{1}{\sigma_j^2}}\, X_i - \mu\right)^2 = E\left(\sum_{i=1}^{n} \frac{\frac{1}{\sigma_i^2}(X_i-\mu)}{\sum_{k=1}^{n}\frac{1}{\sigma_k^2}}\right)^2 = E\left(\sum_{i,j=1}^{n} \frac{\frac{1}{\sigma_i^2}(X_i-\mu)}{\sum_{k=1}^{n}\frac{1}{\sigma_k^2}}\,\frac{\frac{1}{\sigma_j^2}(X_j-\mu)}{\sum_{k=1}^{n}\frac{1}{\sigma_k^2}}\right)$$

Since $X_i's$ are independent, we have $E((X_i-\mu)(X_j-\mu)) = 0$ for $i \neq j$; thus we obtain

$$V(M(\mathbf{X})) = E\left(\sum_{i=1}^{n} \frac{\frac{1}{\sigma_i^4}(X_i-\mu)^2}{\left(\sum_{j=1}^{n}\frac{1}{\sigma_j^2}\right)^2}\right) = \sum_{i=1}^{n} \frac{\frac{1}{\sigma_i^4}\sigma_i^2}{\left(\sum_{j=1}^{n}\frac{1}{\sigma_j^2}\right)^2} = \frac{1}{\sum_{j=1}^{n}\frac{1}{\sigma_j^2}}.$$

Note that this means that the variance of the estimator $M(\mathbf{X})$ is equal $1/n$ times the harmonic mean of the variances of random variables X_i. Recall that the harmonic mean $\mathcal{H}$ of $a_1,\ldots,a_n$ is given by

$$\mathcal{H}(a_1,\ldots,a_n) = \frac{n}{\sum_{i=1}^{n}\frac{1}{a_i}}$$

Can we improve that, i.e., is there an better unbiased estimator with a smaller variance? The answer is no; in this case the ML estimator reaches the lowest possible variance, the information theoretic lower bound for unbiased estimators, also called the Cramer-Rao bound. Before returning on page 9 to problem of optimal aggregation of inconsistent data we include some extended material for the mathematically minded.

Extended material

3 The Cramer – Rao Bound

Theorem 3.1 (The Cramer – Rao Inequality) *Let $\mathbf{X} = (X_1,\ldots,X_n)$ be independent random variables and $f_i(x;\theta)$ their corresponding probability densities depending on a parameter θ in a smooth way, in the sense that*

$$\frac{\partial \ln f_i(x;\theta)}{\partial\theta}$$

exists and is finite for all x such that $f_i(x;\theta) > 0$. Moreover, let $\Theta(\mathbf{X})$ be an unbiased estimator for θ and assume that the variance of $\Theta(\mathbf{X})$ is finite for all θ and that the operations of integration with respect to x and differentiation with respect to θ can be interchanged in the expression for the expectation of Θ:

$$\frac{\partial}{\partial\theta}\int_{\mathbb{R}^n}\Theta(x_1,\ldots,x_n)\prod_{i=1}^{n} f_i(x_i;\theta)\mathrm{d}x_i = \int_{\mathbb{R}^n}\Theta(x_1,\ldots,x_n)\frac{\partial\prod_{i=1}^{n} f_i(x_i;\theta)}{\partial\theta}\mathrm{d}x_1\ldots\mathrm{d}x_n \tag{3.1}$$

Then the variance $\mathrm{var}(\Theta)$ *of* $\Theta(\mathbf{X})$ *satisfies*

$$\mathrm{var}(\Theta) \geq \frac{1}{\sum_{i=1}^{n} E\left[\left(\frac{\partial \ln f_i(x;\theta)}{\partial\theta}\right)^2\right]} \tag{3.2}$$

where each expected value $E\left[\left(\frac{\partial \ln f_i(x;\theta)}{\partial\theta}\right)^2\right]$ is taken with respect to probability density function $f_i(x;\theta)$.

Proof: Since $\Theta(\mathbf{X})$ is an unbiased estimator for θ we obtain

$$E(\Theta(\mathbf{X}-\theta)) = \int_{\mathbb{R}^n}(\Theta(x_1,\ldots,x_n)-\theta)\prod_{i=1}^{n} f_i(x_i,\theta)\mathrm{d}x_i = 0$$

Let us define $\phi(\mathbf{x};\theta) = \prod_{i=1}^{n} f_i(x_i,\theta)$, and $\mathrm{d}\mathbf{x} = \mathrm{d}x_1\ldots\mathrm{d}x_n$; then the above formula simplifies to

$$\int_{\mathbb{R}^n}(\Theta(\mathbf{x})-\theta)\phi(\mathbf{x},\theta)\mathrm{d}\mathbf{x} = 0$$

Differentiating both sides and using our assumption about the correctness of exchange of differentiation and integration we obtain

$$\begin{aligned} 0 =& \frac{\partial}{\partial\theta}\int_{\mathbb{R}^n}(\Theta(\mathbf{x})-\theta)\phi(\mathbf{x},\theta)\mathrm{d}\mathbf{x} = \int_{\mathbb{R}^n}\frac{\partial}{\partial\theta}\left[(\Theta(\mathbf{x})-\theta)\phi(\mathbf{x},\theta)\right]\mathrm{d}\mathbf{x} \\ =& \int_{\mathbb{R}^n}\frac{\partial\left[(\Theta(\mathbf{x})-\theta)\right]}{\partial\theta}\phi(\mathbf{x},\theta)\mathrm{d}\mathbf{x} + \int_{\mathbb{R}^n}(\Theta(\mathbf{x})-\theta)\frac{\partial\phi(\mathbf{x},\theta)}{\partial\theta}\mathrm{d}\mathbf{x} \end{aligned} \tag{3.3}$$

Since $\Theta(\mathbf{x})$ depends only on $\mathbf{x}$ and not on θ, we have

$$\frac{\partial\left[(\Theta(\mathbf{x})-\theta)\right]}{\partial\theta} = -1. \tag{3.4}$$

On the other hand, we have

$$\begin{aligned}\frac{\partial\phi(\mathbf{x},\theta)}{\partial\theta} =& \frac{\partial\prod_{i=1}^{n} f_i(x_i,\theta)}{\partial\theta}\\ =& \sum_{i=1}^{n}\left(\frac{\partial f_i(x_i;\theta)}{\partial\theta}\prod_{j\neq i} f_j(x_j;\theta)\right)\\ =& \sum_{i=1}^{n}\left(\frac{1}{f_i(x_i;\theta)}\frac{\partial f_i(x_i;\theta)}{\partial\theta}\prod_{j=1}^{n} f_j(x_j;\theta)\right)\\ =& \phi(\mathbf{x},\theta)\sum_{i=1}^{n}\frac{\partial\ln f_i(x_i;\theta)}{\partial\theta}\end{aligned} \tag{3.5}$$

Combining (3.3),(3.4) and (3.5) we obtain

$$\begin{aligned}1 &= \int_{\mathbb{R}^n}(\Theta(\mathbf{x})-\theta)\phi(\mathbf{x},\theta)\sum_{i=1}^{n}\frac{\partial\ln f_i(x_i;\theta)}{\partial\theta}\\ &= \int_{\mathbb{R}^n}\left((\Theta(\mathbf{x})-\theta)\sqrt{\phi(\mathbf{x},\theta)}\right)\left(\sqrt{\phi(\mathbf{x},\theta)}\sum_{i=1}^{n}\frac{\partial\ln f_i(x_i;\theta)}{\partial\theta}\right)\mathrm{d}x_1\ldots\mathrm{d}x_n\end{aligned}$$

We now square both sides of this equality and apply the Cauchy Schvarz inequality

$$\left(\int f(x)g(x)\mathrm{d}x\right)^2 \leq \int f(x)^2\mathrm{d}x\cdot\int g(x)^2\mathrm{d}x,$$

thus obtaining

$$1 \leq \int_{\mathbb{R}^n}(\Theta(\mathbf{x})-\theta)^2\phi(\mathbf{x},\theta)\mathrm{d}x_1\ldots\mathrm{d}x_n \quad\cdot\quad \int_{\mathbb{R}^n}\phi(\mathbf{x},\theta)\left(\sum_{i=1}^{n}\frac{\partial\ln f_i(x_i;\theta)}{\partial\theta}\right)^2\mathrm{d}x_1\ldots\mathrm{d}x_n$$

The first integral is just the variance of our estimatr $\Theta(\mathbf{X})$ so we obtain

$$\operatorname{var}(\Theta)\ \cdot\ \int_{\mathbb{R}^n}\phi(\mathbf{x},\theta)\left(\sum_{i=1}^{n}\frac{\partial\ln f_i(x_i;\theta)}{\partial\theta}\right)^2\mathrm{d}x_1\ldots\mathrm{d}x_n \geq 1 \tag{3.6}$$

We now have

$$\begin{aligned}&\int_{\mathbb{R}^n}\phi(\mathbf{x},\theta)\left(\sum_{i=1}^{n}\frac{\partial\ln f_i(x_i;\theta)}{\partial\theta}\right)^2\mathrm{d}x_1\ldots\mathrm{d}x_n =\\ &\quad= \sum_{i,j=1}^{n}\int_{\mathbb{R}^n}\phi(\mathbf{x},\theta)\frac{\partial\ln f_i(x_i;\theta)}{\partial\theta}\frac{\partial\ln f_j(x_j;\theta)}{\partial\theta}\mathrm{d}x_1\ldots\mathrm{d}x_n\\ &\quad= \sum_{i=1}^{n}\int_{\mathbb{R}^n}\phi(\mathbf{x},\theta)\left(\frac{\partial\ln f_i(x_i;\theta)}{\partial\theta}\right)^2\mathrm{d}x_1\ldots\mathrm{d}x_n + \sum_{i\neq j}\int_{\mathbb{R}^n}\phi(\mathbf{x},\theta)\frac{\partial\ln f_i(x_i;\theta)}{\partial\theta}\frac{\partial\ln f_j(x_j;\theta)}{\partial\theta}\mathrm{d}x_1\ldots\mathrm{d}x_n\end{aligned}$$

Since

$$\int_{\mathbb{R}^n} f_i(\mathbf{x},\theta)\mathrm{d}x_i = 1$$

The first sum simplifies to

$$\sum_{i=1}^{n}\int_{\mathbb{R}^n} f_i(\mathbf{x},\theta)\left(\frac{\partial\ln f_i(x_i;\theta)}{\partial\theta}\right)^2\mathrm{d}x_i = \sum_{i=1}^{n}E\left(\frac{\partial\ln f_i(x;\theta)}{\partial\theta}\right)^2,$$

while the second sum simplifies to

$$\sum_{i\neq j}\int_{\mathbb{R}^n} f_i(\mathbf{x},\theta)\frac{\partial \ln f_i(x_i;\theta)}{\partial\theta} f_j(\mathbf{x},\theta)\frac{\partial \ln f_j(x_j;\theta)}{\partial\theta}\mathrm{d}x_i\mathrm{d}x_j =$$

$$\sum_{i\neq j}\int_{\mathbb{R}^n} f_i(\mathbf{x},\theta)\frac{\partial \ln f_i(x_i;\theta)}{\partial\theta}\mathrm{d}x_i\ \cdot\ \int_{\mathbb{R}^n} f_j(\mathbf{x},\theta)\frac{\partial \ln f_j(x_j;\theta)}{\partial\theta}\mathrm{d}x_j =$$

$$\sum_{i\neq j}^{n} E\left(\frac{\partial \ln f_i(x;\theta)}{\partial\theta}\right) E\left(\frac{\partial \ln f_j(x;\theta)}{\partial\theta}\right),$$

Thus, from (3.6) we now obtain

$$\mathrm{var}(\Theta)\ \cdot\ \left(\sum_{i=1}^{n} E\left[\left(\frac{\partial \ln f_i(x;\theta)}{\partial\theta}\right)^2\right] + \sum_{i\neq j}^{n} E\left(\frac{\partial \ln f_i(x;\theta)}{\partial\theta}\right) E\left(\frac{\partial \ln f_j(x;\theta)}{\partial\theta}\right)\right) \geq 1; \tag{3.7}$$

so to prove our theorem it remains to show that second sum above is equal to zero; in fact, we show that each summand is zero. Differentiating with respect to θ both sides of the equations

$$1 = \int_{\mathbb{R}^n} f_i(x_i;\theta)\mathrm{d}x_i$$

we obtain

$$0 = \int_{\mathbb{R}^n}\frac{\partial f_i(x_i;\theta)}{\partial\theta}\mathrm{d}x_i = \int_{\mathbb{R}^n}\frac{1}{f_i(x_i;\theta)}\frac{\partial f_i(x_i;\theta)}{\partial\theta} f_i(x_i;\theta)\mathrm{d}x_i = \int_{\mathbb{R}^n}\frac{\partial \ln f_i(x_i;\theta)}{\partial\theta} f_i(x_i;\theta)\mathrm{d}x_i = E\left(\frac{\partial \ln f_i(x_i;\theta)}{\partial\theta}\right);$$

This, together with (3.7) implies

$$\mathrm{var}(\Theta) \geq \frac{1}{\sum_{i=1}^{n} E\left[\left(\frac{\partial \ln f_i(x;\theta)}{\partial\theta}\right)^2\right]}.$$

□

Let us now see how the above theorem applies to our second example with sensors with unequal variances. In this case the papameter is the mean μ; thus we have

$$f_i(x;\mu) = \frac{1}{\sqrt{2\pi v_i}}\mathrm{e}^{-\frac{(x-\mu)^2}{2v_i}}$$

One can show that the conditions for applicability of the Cramer – Rao bound are satisfied; the interchange of integration and differentiation condition is satisfied due to a quite a general reason: the probability densities functions are continuously differentiable and the expectation integral converges uniformly for all θ.

We now obtain

$$\frac{\partial \ln f_i(x;\mu)}{\partial\mu} = \frac{\partial}{\partial\mu}\left(-\ln\frac{1}{\sqrt{2\pi v_i}} - \frac{(x-\mu)^2}{2v_i}\right) = \frac{x-\mu}{v_i}.$$

Consequently,

$$E\left[\left(\frac{\partial \ln f_i(x;\theta)}{\partial\theta}\right)^2\right] = \int_{-\infty}^{\infty}\frac{(x-\mu)^2}{v_i^2}\frac{1}{\sqrt{2\pi v_i}}\mathrm{e}^{-\frac{(x-\mu)^2}{2v_i}}\mathrm{d}x = \frac{v_i^{-\frac{5}{2}}}{\sqrt{2\pi}}\int_{-\infty}^{\infty} x^2\mathrm{e}^{-\frac{x^2}{2v_i}}\mathrm{d}x \tag{3.8}$$

The last integral is easy to evaluate by integration by parts, by setting $u = x$ and $x\mathrm{e}^{-\frac{x^2}{2v_i}}\mathrm{d}x = v_i\,\mathrm{e}^{-\frac{x^2}{2v_i}}\mathrm{d}\frac{x^2}{2v_i} = \mathrm{d}w$ thus obtaining $w = -v_i\mathrm{e}^{-\frac{x^2}{2v_i}}$ which implies

$$\int_{-\infty}^{\infty} x^2\mathrm{e}^{-\frac{x^2}{2v_i}}\mathrm{d}x = -v_i x\mathrm{e}^{-\frac{x^2}{2v_i}}\Big|_{-\infty}^{+\infty} + v_i\int_{-\infty}^{\infty}\mathrm{e}^{-\frac{x^2}{2v_i}}\mathrm{d}x = \sqrt{2\pi}v_i^{3/2}, \tag{3.9}$$

the last equality following from the fact that

$$\int_{-\infty}^{\infty}\frac{1}{\sqrt{2\pi v_i}}\mathrm{e}^{-\frac{x^2}{2v_i}}\mathrm{d}x = 1.$$

Combining (3.8) and (3.9) we obtain

$$E\left[\left(\frac{\partial \ln f_i(x;\theta)}{\partial\theta}\right)^2\right] = \frac{1}{v_i}$$

This, together with (3.2) implies

$$\operatorname{var}(\Theta) = \frac{1}{\sum_{i=1}^{n}\frac{1}{v_i}} \tag{3.10}$$

Thus, our ML estimator of the measured quantity is the best possible unbiased estimator, in the sense that its variance is minimal possible. We say that such estimator is *efficient.*

(back to the material for all)

We now consider the following practically very important problem.

Problem: In practice variances of the sensors are often unavailable, because they might depend on environmental conditions. Also, sensors are often unattended and might be subject to malicious intrusion by an adversary. So what would be an optimal way to estimate the measured environmental quantity if the variances are unavailable and some sensors might be compromised? Similar problem occurs in many other setups, such as product evaluation, assignment marking, etc, so we formulate a general setup in which we work. Note that in this case the sensors produce real valued outputs, rather than just discrete integers as it was the case in our voting algorithm.

3.1 General Setup

We consider a set of *agents* $A_1, A_2, \ldots, A_n$ making evaluations which they submit to a distinguished aggregating agent which we will call an *authority.*

- **Agents** can be either humans or machines executing some programs. Examples include consumers evaluating products, services or movies and sensors measuring some physical quantity such as temperature.
- **An evaluation** of an agent A_i of an item e_j (or of a physical quantity at an instant j) is denoted by E_{ij}. These evaluations are generally accessible only to the authority to which the agents submit such evaluations.
- **An authority** is an agent which aggregates such, possibly confidential, evaluations into a single **rank** or **rating** assigned to each item evaluated and publish these ratings. Keeping individual evaluations $E_{i,j}$ of agents confidential generally makes the system more robust to malicious misuse, preventing retaliation and blackmail, for example.
- **Our goal** is to provide algorithms for the authority which aggregate these evaluations in an optimal way. This is, of course, a somewhat vague goal, because we did not define what quantity is to be optimised; this will depend on the context of a particular application. However, in general, one would like to have an algorithm with the following features:
  - the algorithm should be robust with respect to outliers. If a small fraction of evaluations E_{ij} are far from some form of a "consensus" of other agents, such evaluations should have very little impact on the final aggregated *ratings* produced by the system;
  - more over, such algorithm should be robust to collusion attacks, where a group of users tries collaboratively to skew the ratings by an orchestrated effort;
  - the algorithm should be "statistically sound"; for example, if individual evaluations E_{ij} provided by any agent A_i are are certain "correct values" plus some zero mean Gaussian noise independent for each item evaluated, then the algorithm should produce an output close to an optimal output, in the sense that the variance of such an output should be close to the Cramer – Rao lower bound.
  - However, *the fact that the error is Gaussian or the standard deviations of errors of agents performing evaluations* ***cannot*** *be an input to the algorithm*, because such kind of information is in practice unavailable; the algorithm must provide an output using only individual measurements of agents.

In general, while the usefulness of a rating method can only be reliably assessed by its performance in practice, we want to examine what kind of supporting evidence one might seek before investing efforts and resources to implement and test such a method by putting it "online". Such evidence can be generally of several kinds:

- mathematical proofs that the algorithm possesses some features; for example, certain statistical behaviour of its outputs, assuming certain (practically justifiable) probability distribution of its inputs, of the type mentioned above. A good feature of such evidence is that it can be obtained prior to investing time and other resources into implementing a system;
- extensive empirical testing of an implementation, using a reasonably large number of available historic samples of such transactions which are reasonably representative of the anticipated transactions for the system being developed;
- when such samples are not available, then by extensive testing using simulated data, making sure that such simulated data is a reasonable, statistically faithful representation of the anticipated transactions.

We now present some algorithms for the above problem and examine their strengths and weaknesses. These algorithms are examples of what is usually called **Iterative Filtering Algorithms**; as we will see they reduce the problem to a computation of a fixed point of a mapping, usually non linear, i.e., to finding a solution to an equation of the form $\boldsymbol{\rho} = F(\boldsymbol{\rho})$. Here $F(\mathbf{x})$ is a mapping from $\mathbb{R}^n$ into $\mathbb{R}^n$ which is sufficiently "well behaved" to allow a proof of existence and uniqueness of such a solution.

All these methods aggregate evaluations E_{ij} of all agents A_i who have evaluated an item e_j into a rating ρ_j of the item e_j using a form of weighted average,

$$\rho_j = \sum_{i:i\to j} w_{ij} E_{ij} \tag{3.11}$$

where $i \to j$ denotes the fact that A_i has evaluated item e_j and has submitted his evaluation E_{ij} to the authority. The weights must satisfy that they are all non-negative, i.e., $w_{ij} \geq 0$ and that for all items j,

$$\sum_{i:i\to j} w_{ij} = 1$$

thus insuring that this sum is a proper average, i.e. that for each fixed j, $1 \leq j \leq M$ it satisfies

$$\min\{E_{ij} \ : \ i \to j\} \leq \sum_{i:i\to j} w_{ij} E_{ij} \leq \max\{E_{ij} \ : \ i \to j\}$$

What should the weights w_i be? The idea behind this technique is to make these weights adaptive so that those agents whose evaluations are closer to the correct ratings get higher weight than those whose evaluation diverge significantly from the correct ratings. But we have no knowledge of what such correct ratings are; in fact, the above weighted average is used precisely to estimate these "true ratings". Even worse, such notion of "true ratings" sometimes makes no sense, for example if the evaluations correspond to one's subjective opinion how entertaining a movie is or even what the "true value" of the stock of a company is.

We replace the "divergence from the true rating" by "divergence from (some kind of) a consensus rating", and rely on the fact that such consensus will reflect opinion of the majority, as well as that the majority do ranking in good faith and reasonably competently.[1]

Note that the proposed strategy is circular: in order to obtain ratings ρ_j of items e_j we want to form a weighted average (3.11) which is such that the weight w_i of each agent A_i reflect how close he is to the ratings which we are trying to assign. Let us denote the collection of all evaluations by $\mathcal{E}$, i.e., let $\mathcal{E} = (E_{ij})_{i,j:i\to j}$. The solution is to represent the weights w_{ij} as functions of the rating to be assigned, i.e., (3.11) should be of the form:

$$\rho_j = \sum_{i:i\to j} w_{ij}(\boldsymbol{\rho}, \mathcal{E}) E_{ij} \quad (1 \leq i \leq n) \tag{3.12}$$

where $w_{ij}(\boldsymbol{\rho})$ are functions which have the property that closer are the evaluations $(E_{i,j})_{j:i\to j}$ of the i^{th} agent to the "consensus values" $\boldsymbol{\rho} = (\rho_1, \ldots, \rho_M)$, the higher the weight $w_{ij}(\boldsymbol{\rho})$ he gets. How close are the evaluations $(E_{i,j})_{j:i\to j}$ of the i^{th} agent to the "consensus values" $\boldsymbol{\rho} = (\rho_1, \ldots, \rho_M)$ will be used to assess the *trustworthiness* τ_i of an agent, and thus the weights w_{ij} will depend on such trustworthiness. One might wonder why the weight of an agent i, when evaluating an item j, should depend on j. Intuitively, one feels that such a weight should only depend on the agent's overall trustworthiness. In fact this is essentially true, except that the set of agents which have evaluated an item j might differ for different items, so an "absolute trustworthiness" of an agent must be appropriately scaled to ensure that weights of all agents evaluating an item sum up to 1. Thus, in general,

$$w_{ij}(\boldsymbol{\rho}, \mathcal{E}) = \frac{\tau_i(\boldsymbol{\rho}, \mathcal{E})}{\sum_{r\to j} \tau_r(\boldsymbol{\rho}, \mathcal{E})},$$

[1] Later we will look into cases when a group of colluding agents manages to completely "hijack" the rating of an item and skew it far away from its "reasonable value".

and (3.19) will be of the form

$$\rho_j = \sum_{i:i\to j} \frac{\tau_i(\boldsymbol{\rho},\mathcal{E})}{\sum_{r\to j}\tau_r(\boldsymbol{\rho},\mathcal{E})} E_{ij} \quad (1 \le i \le n) \tag{3.13}$$

We now see that, putting together all of equations (3.19) for all agents A_i, $1 \le i \le n$, into a single vector equation, we obtain

$$\boldsymbol{\rho} = \left(\sum_{i:i\to j} w_{ij}(\boldsymbol{\rho},\mathcal{E})\, E_{ij}\right)_{1\le j\le M} = \left(\sum_{i:i\to j} \frac{\tau_i(\boldsymbol{\rho},\mathcal{E})}{\sum_{r\to j}\tau_r(\boldsymbol{\rho},\mathcal{E})}\, E_{ij}\right)_{1\le j\le M} \tag{3.14}$$

i.e., that we are looking for a fixed point of the mapping $F(\mathbf{x}) : \mathbb{R}^M \to \mathbb{R}^M$ defined by

$$F(\mathbf{x}) = \left(\sum_{i:i\to j} w_{ij}(\mathbf{x},\mathcal{E}) E_{ij}\right)_{1\le j\le M}$$

We now consider several possibilities of how to define such functions w.

3.2 Case study #1: P. Laureti, L. Moret, Y.-C. Zhang and Y.-K. Yu model

We start with a model from *P. Laureti, L. Moret, Y.-C. Zhang and Y.-K. Yu: Information filtering via Iterative Refinement*, `http://arxiv.org/pdf/physics/0608166v1.pdf` (see also the references at the end), and consider the following example.

Example 1: Assume that you are a conference chair; you have received quite a few submissions so you have a number of referees to help you select the best papers to be presented at the conference.

- each submission has been reviewed by several referees and every referee has reviewed several papers;
- some of the referees might have been unreasonably harsh with their marks (no one is as smart as they are!);
- some might have been sloppy, barely having looked at the papers so they are likely to have made large random errors;
- worse, some of the referees might have colluded in order to promote the papers of their friends and trash the papers of those against whom they might hold grudges. (Note that this is highly hypothetical; academics always have the highest standards of behaviour and holding grudges against their colleagues is unheard of!)

Question: How should you aggregate the referee's scores and decide which papers to accept in the fairest possible way?

To make things more precise, lets assume that we have R many referees marking P many papers. We denote by E_{rp} the mark which referee r has assigned to paper p (providing that referee r has evaluated paper p).

The aim now is to obtain the aggregate ratings $\boldsymbol{\rho} = (\rho_p \;:\; 1 \le p \le P)$ of all submitted papers. As in our voting algorithm, we will produce such simultaneously with estimates of trustworthiness of the referees, $\boldsymbol{\tau} = (\tau_r \;:\; 1 \le r \le R)$. Both $\boldsymbol{\rho}$ and $\boldsymbol{\tau}$ are obtained simultaneously via a single recursive procedure. As with our voting algorithm, we start by giving equal trustworthiness to all referees:

$$\tau_r^{(0)} = 1;$$

Consequently, the initial weights $w_{rj}^{(0)}$ will be given by

$$w_{rj}^{(0)} = \frac{\tau_r^{(0)}}{\sum_{i:i\to j}\tau_i^{(0)}} = \frac{1}{|\{i : i \to j\}|}$$

and the initial estimate of ratings of items will be given by

$$\rho_j^{(0)} = \sum_{r:r\to j} \frac{\tau_r}{\sum_{i:i\to j}\tau_i} E_{rj} = \sum_{r:r\to j} \frac{1}{|\{i : i \to j\}|} E_{rj}.$$

Since $|\{i : i \to j\}|$ is just the number of referees who have referred paper j, the value of $\rho_j^{(0)}$ is just a simple mean of all marks of the referees who have refereed paper j.

Having such an initial, rough estimate of the marks $\rho_j^{(0)}$ of each paper j we can now turn the table in the same way as we did in our voting algorithm and evaluate how trustworthy each referee is, by estimating how close he was to the fair marks of the papers. Since we do not have such objective marks, we make an approximate evaluation by estimating how close each referee was to the rough, initial approximation $\rho_j^{(0)}$ of the marks, in other words we compute the Euclidean distance d_r of the marks of each referee r to such approximate marks $\boldsymbol{\rho}^{(0)}$:

$$d_\rho^{(0)} = \sqrt{\sum_{j:r\to j} (E_{rj} - \rho_j^{(0)})^2}.$$

We can now obtain a new estimate of trustworthiness of a rater r, as

$$\tau_r^{(1)} = F(d_r^{(0)}),$$

where $F(x)$ is a decreasing function of x, because the larger is the distance between the marks by a referee and the approximate marks $\boldsymbol{\rho}^{(0)}$, the smaller trustworthiness we want to give to such a referee. What is the right choice for such decreasing function $F(x)$? This turns out to be a very tricky question which we will discuss later; we first describe our iterative algorithm in general with a "generic", unspecified but monotonically decreasing function $F(x)$.

Assuming we have obtained vectors $\boldsymbol{\rho}^{(n)}$, we can compute the Euclidean distances

$$d_\rho^{(n)} = \sqrt{\sum_{j:r\to j} (E_{rj} - \rho_j^{(n)})^2},$$

and the corresponding trustworthiness ranks of each referee, i.e., we can obtain

$$\tau_r^{(n+1)} = F(d_r^{(n)}).$$

We can now compute the new weights $w_{rj}^{(n)}$ and new marks of each paper via

$$\begin{aligned} w_{ij}^{(n+1)} &= \frac{\tau_r^{(n+1)}}{\sum_{i:i\to j} \tau_i^{(n+1)}}; \\ \rho_j^{(n+1)} &= \sum_{r:r\to j} \frac{\tau_r^{(n+1)}}{\sum_{i:i\to j} \tau_i^{(n+1)}} E_{rj}. \end{aligned}$$

In this particular paper the authors choose $F(x) = \frac{1}{x^2}$; thus we have

$$\begin{aligned} \tau_r^{(n+1)} &= \frac{1}{(d_r^{(n)})^2} = \frac{1}{\sum_{j\,:r\to j} (E_{rj} - \rho_j^{(n)})^2}; \\ w_{ij}^{(n+1)} &= \frac{\tau_i^{(n+1)}}{\sum_{k:\,k\to j} \tau_i^{(n+1)}} = \frac{\frac{1}{(d_i^{(n)})^2}}{\sum_{m:m\to j} \frac{1}{(d_m^{(n)})^2}} = \frac{\frac{1}{\sum_{p\,:\,i\to p}(E_{ip} - \rho_p^{(n)})^2}}{\sum_{m:m\to j} \frac{1}{\sum_{q\,:\,m\to q}(E_{mq} - \rho_q^{(n)})^2}}; \\ \rho_j^{(n+1)} &= \sum_{i\,:\,i\to j} w_{ij}^{(n+1)} E_{ij}. \end{aligned}$$

Thus we are looking for a fixed point $\boldsymbol{\rho}$, i.e., a vector $\boldsymbol{\rho}$ such that for each j and $\rho_j = (\boldsymbol{\rho})_j$ we have

$$\rho_j = \sum_{k\,:\,k\to j} \frac{\frac{1}{\sum_{p\,:\,k\to p}(E_{kp} - \rho_p)^2}}{\sum_m \frac{1}{\sum_{q\,:\,m\to q}(E_{mq} - \rho_q)^2}} E_{kj} \tag{3.15}$$

Note that the above iterative procedure has a convenient explanation: at the stage of iteration $n + 1$ we use the previously obtained values $\rho_j^{(n)}$ to estimate the variances v_r of the referees, as

$$v_r^{(n)} = \sum_{j\,:\,r\to j} (E_{ip} - \rho_p)^2,$$

and then, with such obtained approximations of the variances of the referees we obtain the new estimate of the "true marks" of each paper using a corresponding "approximate Maximum Likelihood Estimation" of the new marks, i.e., as

$$\rho_j^{(n+1)} = \sum_{k\,:\,k\to j} \frac{\frac{1}{v_k^{(n)}}}{\sum_m \frac{1}{v_m^{(n)}}} E_{kj}, \tag{3.16}$$

i.e., as an ML estimate with $v_k^{(n)}$ in place of the true variances v_k of the referees.

How well does the above algorithm perform? On the class website you will find Mathematica implementations of several Iterative Filtering algorithms which we will soon run in class trying to evaluate their performance. As you will see, one of the problems with the above method is that the function $F(x) = \frac{1}{x^2}$ has a pole at zero. If, in the course of iteration one of the distances $d_k^2 = \sum_{p\,:\,k\to p}(E_{kp} - \rho_p)^2$ gets very small, the value of $1/d_p^2$ will become very large and in this way the weights w_{kj} for referee k will be very close to one, and the weights corresponding to all other referees will be close to zero. As a result, in just a few rounds of iteration the marks ρ_j will become essentially equal to the marks E_{kj} of referee k. Thus, marks of each referee in our iterative procedure act as *attractors*, often causing the mark estimates to converge to marks of one of the referees. This is most likely (but not necessarily only then) happens if the matrix of all evaluations is sparse (i.e., every rater has rated only avery small number of all items).

One can try to fix such a problem via regularisation, by choosing $F(x) = 1/(x^2 + \varepsilon)$ instead of $F(x) = 1/x^2$, where ε is a small positive constant whose role is to prevent the denominator from vanishing. Unfortunately, this does not work vey well: it turns out that either ε is chosen too small to ensure stability of the recursive procedure, or it is too large in the sense that the values of ρ_i obtained via such iterative procedure do not differ too much from a simple arithmetic mean.

Another way of deling with this problem is to choose for $F(x)$ a better behaved function which has no poles. This was attempted in the model we describe next, which is also implemented in Mathematica files on our website.

3.3 Case study #2: the Kerchove-Dooren model

We now go through the paper "C. de Kerchove, P. VanDooren: *Iterative filtering for a Dynamical Reputation System*"`http://perso.uclouvain.be/paul.vandooren/publications/deKerchoveV07.pdf`".

The setup of this paper is quite similar to the setup of the previous paper, with some differences. We again have for each object j its rating ρ_j given by

$$\rho_j = \sum_{i:i\to j} w_{ij} E_{ij} \quad (1 \le i \le n), \qquad \text{(equation (1) in their paper)} \tag{3.17}$$

with the weights w_{ij} of the form

$$w_{ij} = \frac{\tau_{ij}}{\sum_{k:k\to j} \tau_{kj}} \qquad \text{(equation (2) in the paper)} \tag{3.18}$$

where the quantities τ_{ij} reflect the "trustworthiness" of agent i when evaluating item j. Thus, unlike the previous algorithm, in the model presented in this paper there is no single, overall trustworthiness τ_i of an agent i, but rather, for every item j we have a trustworthiness τ_{ij} of agent i when evaluating item j, i.e., now the trustworthiness depends also on the item being evaluated, and not only the evaluating agent i.

To obtain τ_{ij} we first define an "average distance" function of the i^{th} agent's evaluations $(E_{ij})_{1\le j\le M}$ to a ranking vector $\boldsymbol{\rho} = (\rho_j)_{1\le j\le M}$, as the square of the RMS (Root Mean Square) value of his "error". The authors call such quantity "belief divergence".Thus, assume that m_i is the number of items the i^{th} agent has evaluated. Then we define such belief divergence of the i^{th} agent as

$$d_i = \frac{1}{m_i} \sum_{j:i\to j} (E_{ij} - \rho_j)^2 \qquad \text{(equation (3) in the paper)} \tag{3.19}$$

Note that the Euclidean distance between $\boldsymbol{\rho} = (\rho_j)_{1\le j\le M}$ and $(E_{ij})_{1\le j\le M}$ is $\sqrt{\sum_{j:i\to j}(E_{ij} - \rho_j)^2} = \sqrt{m_i\, d_i}$, while the RMS of the "error" of his evaluations $(E_{ij})_{1\le j\le M}$, taking $\boldsymbol{\rho} = (\rho_j)_{1\le j\le M}$ as "the correct values", is $\sqrt{d_i}$. The corresponding trust values τ_{ij} are now defined as

$$\tau_{ij} = c_j - d_i$$

where c_i are some constants which correspond to each item j and chosen such that $c_j - d_i \geq 0$ for every agent i evaluating item j. Clearly, again the trustworthiness of each agent is *inversely dependent* on his belief divergence because indeed τ_{ij} decreases if d_i increases; however, this time the functions $F_j(x) = c_j - x$ which correspond to (a single) function $F(x) = 1/x^2$ from the first case study no longer have a pole at $x = 0$, and consequently the values E_{ij} provided by an individual agent i no longer act as attractors for the corresponding recursive procedure to be described below.

How large should such constants c_i be? We have to ensure that all the weights w_{ij} are non-negative, and from formula (3.18) we see that this will be the case if all τ_{ij} are also positive, which happens just in case $c_j \geq d_i$. Note that d_i is actually a function of the rank vector $\boldsymbol{\rho}$ (the evaluation values E_{ij} are constant while we search for an appropriate rank vector $\boldsymbol{\rho}$); thus, such inequality should hold at least for all $\mathbf{x}$ in a neighbourhood around the solution $\boldsymbol{\rho}$ in which we will be searching for such a solution; we will come back to this issue later, after we describe the algorithm.

The algorithm in the section 2.3 of the paper is an iterative procedure which finds a fixed point, even though the authors do not explicitly say this.

The algorithm starts by assigning to each agent i the same trust rank $\tau_{ij}^{(0)}$ for every item j, i.e., $\tau_{ij}^{(0)} = 1$. We then compute the corresponding values of the weights obtained from (3.18) as

$$w_{ij}^{(0)} = \frac{\tau_{ij}^{(0)}}{\sum_{k:k\rightarrow j} \tau_{kj}^{(0)}} \tag{3.20}$$

Let n_j be the number of agents who have evaluated item j, thus

$$n_j = |\{i \,:\, i \rightarrow j\}|;$$

since all τ_{ij} are set to 1, we get that

$$w_{ij}^{(0)} = \frac{1}{n_j} \tag{3.21}$$

Thus, from (3.19) we get that the initial assessment of the ranking of the j^{th} item is given by

$$\rho_j^{(0)} = \sum_{i:i\rightarrow j} \frac{1}{n_j} E_{ij} = \frac{\sum_{i:i\rightarrow j} E_{ij}}{n_j}, \qquad (1 \leq j \leq M). \tag{3.22}$$

i.e., just like in the first case study, it is simply the arithmetical mean of evaluations of all agents who evaluated item j.

As in the previous case, we take such ranks as a temporary "true ratins" and compute the belief divergences of each evaluating agent, i.e.,

$$d_i^{(0)} = \frac{1}{m_i} \sum_{j:i\rightarrow j} (E_{ij} - \rho_j^{(0)})^2 \tag{3.23}$$

and compute the corresponding new trust ranks of agents and their corresponding weights

$$\tau_{ij}^{(1)} = c_j - d_i^{(0)}; \qquad w_{ij}^{(1)} = \frac{\tau_{ij}^{(1)}}{\sum_{k:k\rightarrow j} \tau_{kj}^{(1)}}.$$

The idea behind is that, even though the simple averages $\boldsymbol{\rho}^{\,(0)}$ might be skewed by the outliers, the worst outliers will be farthest from such averages and will consequently receive lower new trustworthiness ranks $\tau_{ij}^{(1)}$ and thus also lower corresponding weights $w_{ij}^{(1)}$. Consequently, the new ranks $\boldsymbol{\rho}^{\,(1)}$ obtained as weighted averages with these new weights will be less skewed by the outliers than the simple averages $\boldsymbol{\rho}^{\,(0)}$.

We now continue the recursion always computing the new belief divergences, new trust ranks of agents, new corresponding weights and the new rankings of items from previously computed ranking $\boldsymbol{\rho}^{\,(n)}$ by

$$\begin{aligned}
d_i^{(n)} &= \frac{1}{m_i} \sum_{j:i\rightarrow j} (E_{ij} - \rho_j^{(n)})^2; \\
\tau_{ij}^{(n+1)} &= c_j - d_i^{(n)}; \\
w_{ij}^{(n+1)} &= \frac{\tau_{ij}^{(n+1)}}{\sum_{k:k\rightarrow j} \tau_{kj}^{(n+1)}}; \\
\rho_j^{(n+1)} &= \sum_{i:i\rightarrow j} w_{ij}^{(n+1)} E_{ij}.
\end{aligned}$$

In each round of recursion the outliers will be increasingly marginalised by obtaining progressively lower weights. When should such a recursion terminate? We can set up a small threshold value ε and stop recursion when the new ranks and the previous ranks differ less then ε, i.e. when for all $1 \leq j \leq M$

$$|\rho_j^{(n+1)} - \rho_j^{(n)}| < \varepsilon$$

Since eventually for sufficiently large n we have

$$\rho_j^{(n)} \approx \rho_j^{(n+1)} = \sum_{i:i\to j} w_{ij}^{(n+1)} E_{ij}$$

by expressing $w_{ij}^{(n+1)}$ in terms of $\rho_j^{(n)}$ using the recursion equations we obtain that

$$\rho_j^{(n)} \approx \sum_{i:i\to j} \frac{\tau_{ij}^{(n+1)}}{\sum_{k:k\to j} \tau_{kj}^{(n+1)}} E_{ij} = \sum_{i:i\to j} \frac{c_j - d_i^{(n)}}{\sum_{k:k\to j}(c_j - d_k^{(n)})} E_{ij} =$$

$$\sum_{i:i\to j} \frac{c_j - \frac{1}{m_i}\sum_{p:i\to p}(E_{ip} - \rho_p^{(n)})^2}{\sum_{k:k\to j}\left(c_j - \frac{1}{m_k}\sum_{l:k\to l}(E_{kl} - \rho_l^{(n)})^2\right)} E_{ij}$$

Thus, the recursion will stop when $\rho_j^{(n)}$ is close to the solution $\boldsymbol{\rho} = (\rho_j)_{1\leq j\leq M}$ of the system of equations

$$\rho_j = \sum_{i:i\to j} \frac{c_j - \frac{1}{m_i}\sum_{p:i\to p}(E_{ip} - \rho_p)^2}{\sum_{k:k\to j}\left(c_j - \frac{1}{m_k}\sum_{l:k\to l}(E_{kl} - \rho_l)^2\right)} E_{ij}, \qquad (1 \leq j \leq M) \tag{3.24}$$

or, putting these equations in a single vector equation, when

$$\boldsymbol{\rho} = \left(\sum_{i:i\to j} \frac{c_j - \frac{1}{m_i}\sum_{p:i\to p}(E_{ip} - (\boldsymbol{\rho})_p)^2}{\sum_{k:k\to j}\left(c_j - \frac{1}{m_k}\sum_{l:k\to l}(E_{kl} - (\boldsymbol{\rho})_l)^2\right)} E_{ij}\right)_{1\leq j\leq M} \tag{3.25}$$

with $(\boldsymbol{\rho})_k$ denoting the projection, i.e., the k^{th} coordinate of $\boldsymbol{\rho}$. We now see that the algorithm stops when we obtain a sufficiently close approximation of the *fixed point* of the (non-linear) mapping $F : \mathbb{R}^M \to \mathbb{R}^M$ defined for all vectors $\mathbf{x} = (x_j)_{1\leq j\leq M}$ by the equation

$$F(\mathbf{x}) = \left(\sum_{i:i\to j} \frac{c_j - \frac{1}{m_i}\sum_{p:i\to p}(E_{ip} - (\mathbf{x})_p)^2}{\sum_{k:k\to j}\left(c_j - \frac{1}{m_k}\sum_{l:k\to l}(E_{kl} - (\mathbf{x})_l)^2\right)} E_{ij}\right)_{1\leq j\leq M} \tag{3.26}$$

How well does this form of iterative filtering perform compared with the previous one with the reciprocal function? At the class website you will find a Mathematica file which we will run in class and which also contains an iterative filtering algorithm designed at UNSW and which beats both of the algorithms presented, especially in collusion attacks. Before presenting two such algorithms developed at UNSW we include an analysis of the above algorithm from the original paper cited. The material for all students is continued on page 20.

Extended Material

Analysis provided in the paper and presented below, while not entirely correct (as you will see) is a good example of the techniques one might use to analyse iterative algorithms of this kind. It can be your major project to provide such analysis for an algorithm designed at UNSW which we will present later and for which such an analysis has not been done (i.e., a a proof of existence of the fixed point, its uniqueness and a proof of convergence of the corresponding iterative procedure).

We now have to show that there always exists a unique such fixed point, that our algorithm converges to the fixed point, and that the ranks defined by such fix point have properties which justify their use. The authors propose to show that the fixed point condition represents a necessary condition for a corresponding function to have an extremum at that point. Thus, let us consider all trust ratings

$$\tau_{ij}(\boldsymbol{\rho}) = c_j - \frac{1}{m_i}\sum_{p\,:\,i\to p}(E_{ip} - (\boldsymbol{\rho})_p)^2$$

and form the following (sparse) trust matrix with non-zero entries τ_{ij} only for i, j such that agent i has evaluated item j, i.e., such that $i \to j$, which looks something like this:

$$G(\boldsymbol{\rho}) = \begin{pmatrix} 0 & \dots & 0 & \dots & \tau_{1j}(\boldsymbol{\rho}) & \dots & 0 & \dots & 0 \\ \vdots & & \vdots & & \vdots & & \vdots & & \vdots \\ 0 & \dots & \tau_{ij_1}(\boldsymbol{\rho}) & \dots & 0 & \dots & \tau_{ij_2}(\boldsymbol{\rho}) & \dots & 0 \\ \vdots & & \vdots & & \vdots & & \vdots & & \vdots \\ 0 & \dots & 0 & \dots & \tau_{nj}(\boldsymbol{\rho}) & \dots & 0 & \dots & \tau_{nM}(\boldsymbol{\rho}) \end{pmatrix}$$

The *Frobenius norm* $\|G\|_2$ of a matrix is just the square root of the sum of the squares of all of its elements; thus in this case

$$g(\boldsymbol{\rho}) = \|G(\boldsymbol{\rho})\|_2^2 = \sum_{1 \le i \le n} \sum_{j\,:\,i \to j} \tau_{ij}^2(\boldsymbol{\rho})$$

The authors now claim that the solution to (3.25) corresponds to $\boldsymbol{\rho}$ which maximises the Frobenius norm $g(\boldsymbol{\rho})$ of $G_T(\boldsymbol{\rho})$. To find the extremal points of this function we compute its gradient

$$\nabla g(\boldsymbol{\rho}) = \left(\frac{\partial}{\partial \rho_1} g(\boldsymbol{\rho}), \dots, \frac{\partial}{\partial \rho_M} g(\boldsymbol{\rho}) \right)$$

Note that a trust expression $\tau_{ij}(\boldsymbol{\rho}) = c_j - \frac{1}{m_i} \sum_{p:i \to p} (E_{ip} - (\boldsymbol{\rho})_p)^2$ depends on ρ_k just in case $i \to k$; thus

$$\begin{aligned} \frac{\partial}{\partial \rho_k} g(\boldsymbol{\rho}) = & \sum_{i=1}^{n} \sum_{j\,:\,i \to j} \frac{\partial}{\partial \rho_k} \tau_{ij}^2(\boldsymbol{\rho}) = \sum_{i\,:\,i \to k} \sum_{j\,:\,i \to j} \frac{\partial}{\partial \rho_k} \left(c_j - \frac{1}{m_i} \sum_{p:i \to p} (E_{ip} - (\boldsymbol{\rho})_p)^2 \right)^2 = \\ & \sum_{i\,:\,i \to k} \sum_{j\,:\,i \to j} \left(c_j - \frac{1}{m_i} \sum_{p:i \to p} (E_{ip} - (\boldsymbol{\rho})_p)^2 \right) 2 \left(-\frac{1}{m_i} \right) \sum_{p:i \to p} \frac{\partial}{\partial \rho_k} (E_{ip} - (\boldsymbol{\rho})_p)^2; \end{aligned}$$

Clearly, partials $\frac{\partial}{\partial \rho_k} (E_{ip} - (\boldsymbol{\rho})_p)^2$ are all zero except for $p = k$, in which case $\frac{\partial}{\partial \rho_k} (E_{ik} - \rho_k)^2 = -2(E_{ik} - \rho_k)$. Thus we get:

$$\begin{aligned} \frac{\partial}{\partial \rho_k} g(\boldsymbol{\rho}) = & -2 \sum_{i\,:\,i \to k} \sum_{j\,:\,i \to j} \left(c_j - \frac{1}{m_i} \sum_{p:i \to p} (E_{ip} - \rho_p)^2 \right) \frac{1}{m_i} \frac{\partial}{\partial \rho_k} (E_{ik} - \rho_k)^2 = \\ & 4 \sum_{i\,:\,i \to k} \sum_{j\,:\,i \to j} \left(c_j - \frac{1}{m_i} \sum_{p:i \to p} (E_{ip} - \rho_p)^2 \right) \frac{1}{m_i} (E_{ik} - \rho_k) = \\ & 4 \sum_{i\,:\,i \to k} \frac{1}{m_i} (E_{ik} - \rho_k) \sum_{j\,:\,i \to j} (c_j - d_i(\boldsymbol{\rho})) = \\ & 4 \sum_{i\,:\,i \to k} (E_{ik} - \rho_k) \left(\frac{\sum_{j\,:\,i \to j} c_j}{m_i} - \frac{\sum_{j\,:\,i \to j} d_i(\boldsymbol{\rho})}{m_i} \right) = \\ & 4 \sum_{i\,:\,i \to k} E_{ik} \left(\frac{\sum_{j\,:\,i \to j} c_j}{m_i} - d_i(\boldsymbol{\rho}) \right) - 4\, \rho_k \sum_{i\,:\,i \to k} \left(\frac{\sum_{j\,:\,i \to j} c_j}{m_i} - \frac{m_i\, d_i(\boldsymbol{\rho})}{m_i} \right) \\ & 4 \sum_{i\,:\,i \to k} E_{ik} \left(\frac{\sum_{j\,:\,i \to j} c_j}{m_i} - d_i(\boldsymbol{\rho}) \right) - 4\, \rho_k \sum_{i\,:\,i \to k} \left(\frac{\sum_{j\,:\,i \to j} c_j}{m_i} - d_i(\boldsymbol{\rho}) \right) \end{aligned}$$

Note that in the above we used the fact that in the sum $\sum_{j\,:\,i \to j} d_i(\boldsymbol{\rho})$ the summand $d_i(\boldsymbol{\rho})$ does not depend on j; thus, this sum is just the number of elements in the set $\{j : i \to i\}$ times $d_i(\boldsymbol{\rho})$, i.e., $m_i\, d_i(\boldsymbol{\rho})$ in our notation.

We now see that <u>**if**</u> all the constants c_j are equal to some c_0, then

$$\sum_{j\,:\,i \to j} c_j = m_i c_0$$

and also $\tau_{ik}(\boldsymbol{\rho}) = \tau_{ij}(\boldsymbol{\rho})$ for all i, k; thus,

$$\frac{\partial}{\partial \rho_k} g(\boldsymbol{\rho}) = 4 \sum_{i\,:\,i \to k} \tau_{ik}(\boldsymbol{\rho})\, E_{ik} - 4\, \rho_k \sum_{i\,:\,i \to k} \tau_{ik}(\boldsymbol{\rho}) \tag{3.27}$$

which implies $\frac{\partial}{\partial \rho_k} g(\boldsymbol{\rho}) = 0$ just in case $\sum_{\, i\,:\, i \to k} \tau_{ik}\, E_{ik} = \rho_k \sum_{\, i\,:\, i \to k} \tau_{ik}$; changing the second index of sumation (a "dummy variable", as it is called) to avoid confusion, we obtain

$$\frac{\partial}{\partial \rho_k} g(\boldsymbol{\rho}) = 0 \qquad \Leftrightarrow \qquad \rho_k = \sum_{i\,:\, i \to k} \frac{\tau_{ik}}{\sum_{p\,:\, p \to k} \tau_{pk}}\, E_{ik}$$

Thus, every solution of $\boldsymbol{\rho} = F(\boldsymbol{\rho})$ corresponds to a point where $\nabla g(\boldsymbol{\rho}) = 0$; such points where the gradient of a function $g(\boldsymbol{\rho})$ is zero are called the *stationary points* of $g(\boldsymbol{\rho})$.

However, the above calculation holds **only if** all c_j are equal, so the statement on the top of page 5 of the paper:

> Let r^* represents that solution and for the sake of simplicity, let parameters c_j be equals to a same constant c_0.

is certainly an "understatement", because it is not about "simplicity" but without this assumption the theorem does not hold. We will later explain why the authors needed to go to trouble to introduce different constants at all.

As with one variable case, to conclude that such a point is a maximum of $g(\boldsymbol{\rho})$ one has to perform a second order derivatives test: a sufficient condition that a stationary point is a maximum is to compute the matrix of second order partials $\left(\frac{\partial^2}{\partial \rho_i \partial \rho_j} g(\boldsymbol{\rho})\right)_{1 \leq i,j \leq M}$ and show that all of it eigenvalues are negative. However, to obtain that there is a unique stationary point in this particular case, one could show that $g(\boldsymbol{\rho})$ is quasi-concave (explained later); authors skip this unpleasant technicality and we will also give them the benefit of the doubt :-)

The proof that there is a unique solution to $\boldsymbol{\rho} = F(\boldsymbol{\rho})$ which is a maximum of $g(\boldsymbol{\rho})$, as we will see a bit later, will justify the convergence of our algorithm; but before we proceed with this aspect, let us see why we might expect that such a solution should have a good practical performance just from the mathematical features of such a solution.

Assume that our n agents are sensors which provide "true values" plus some white noise, i.e., that their readings have all zero mean Gaussian independently distributed error. Assume that the variances of sensors are $\sigma_1^2, \ldots, \sigma_n^2$; assume also that the correct values of the quantity measured at instants of time $t_1, \ldots, t_M$ are $\rho_1, \ldots, \rho_M$; then the values E_{ij} are of the form $E_{ij} = \mathcal{N}(\rho_j, \sigma_i)$. Not all sensors make a measurement at each of M instants; as before, we write $i \to j$ if i^{th} sensor makes a measurement at instant j.

Let $E^i = (E_{ij})_{1 \leq j \leq M}$ and assume that for each sensor i we have the corresponding readings E^i as well as its variance σ_i^2 and would like to estimate the expected values ρ_j. Let us compute *the likelihood* $\mathcal{L}(E^i; \boldsymbol{\rho})$ of getting the readings E^i of the i^{th} sensor in terms of $\boldsymbol{\rho}$. Since the errors are Gaussian and independently distributed, we get that for every fixed i we have

$$\mathcal{L}(E^i; \boldsymbol{\rho}) = \prod_{j\,:\, i \to j} \frac{1}{\sqrt{2\pi\sigma_i^2}}\, e^{-\frac{1}{2}\frac{(E_{ij} - \rho_j)^2}{\sigma_i^2}} \tag{3.28}$$

Since products are cumbersome to work with, we take the log of both sides and use the fact that, by definition, $m_i = |\{j \,:\, i \to j\}|$:

$$\begin{aligned} \ln \mathcal{L}(E^i; \boldsymbol{\rho}) = \sum_{j\,:\, i \to j} \ln\left(\frac{1}{\sqrt{2\pi\sigma_i^2}}\right) - \frac{1}{2} \sum_{j\,:\, i \to j} \frac{(E_{ij} - \rho_j)^2}{\sigma_i^2} = \\ - \frac{1}{2} m_i \ln\left(2\pi\sigma_i^2\right) - \frac{1}{2\sigma_i^2} \sum_{j\,:\, i \to j} (E_{ij} - \rho_j)^2 \end{aligned}$$

by multiplying both sides by $2\sigma_i^2$, dividing by m_i and adding and subtracting c_l we get

$$\begin{aligned} \frac{2\sigma_i^2}{m_i} \ln \mathcal{L}(E^i; \boldsymbol{\rho}) = - \sigma_i^2 \ln\left(2\pi\sigma_i^2\right) - c_l + \left(c_l - \frac{1}{m_i} \sum_{j\,:\, i \to j} (E_{ij} - \rho_j)^2 \right) = \\ - \sigma_i^2 \ln\left(2\pi\sigma_i^2\right) - c_l + \tau_{il} \end{aligned}$$

i.e., changing index l into a more convenient j:

$$\tau_{ij} = \frac{2\sigma_i^2}{m_i} \ln \mathcal{L}(E^i; \boldsymbol{\rho}) + \sigma_i^2 \ln\left(2\pi\sigma_i^2\right) + c_j \tag{3.29}$$

The authors now suggest that the constant c_j should be chosen so that $\ln\left(2\pi\sigma_i^2\right) + \frac{c_j}{2\sigma_i^2} = 0$, i.e.

$$c_j = -2\sigma_i^2 \ln\left(2\pi\sigma_i^2\right)$$

Since c_j depends on which item j is evaluated and σ_i on the evaluator i, we see that again the calculation goes through only in case all constants c_j are equal, i.e., $c_i = c_0$ for some c_0, but also only in case all evaluator have the same variance $\sigma_i^2 = \sigma^2$. Moreover, since the constant c_0 must be positive, from

$$c_0 = -2\sigma^2 \ln\left(2\pi\sigma^2\right)$$

we get that the variance of the evaluators must be small, in order for the logarithm to be negative, namely that $2\pi\sigma^2 < 1$, which holds just in case

$$\sigma^2 < \frac{1}{2\pi}$$

Thus, again, the above calculations go through only under very restricted conditions, which are by no means just for "the sake of simplicity". Granting that such conditions are satisfied, we get

$$\tau_{ij} = \frac{2\sigma^2}{m_i} \ln \mathcal{L}(E^i; \boldsymbol{\rho})$$

Thus, maximizing the Frobenius norm of the trust matrix amounts to maximizing the following function of the likelihoods $\mathcal{L}(E^i; \boldsymbol{\rho})$:

$$\begin{aligned} \|G(\boldsymbol{\rho})\|^2 = \sum_{1\leq i\leq n} \; \sum_{j\,:\, i\to j} \tau_{ij}^2(\boldsymbol{\rho}) = \\ \sum_{1\leq i\leq n} m_i \frac{4\sigma^4}{m_i^2} (\ln \mathcal{L}(E^i; \boldsymbol{\rho}))^2 = \\ 4\sigma^4 \sum_{1\leq i\leq n} \frac{(\ln \mathcal{L}(E^i; \boldsymbol{\rho}))^2}{m_i} \end{aligned}$$

Since m_i is equal to the number of items rated by agent i the last expression is a multiple of the square of the quadratic mean of the log-likelihoods of the event to get ratings E_{ij} assuming that the correct values are $\boldsymbol{\rho}$, and consequently, maximising $\|G(\boldsymbol{\rho})\|^2$ is maximising such a quadratic mean. While this is NOT the usual MLE, it is intuitively a reasonably meaningful quantity which asserts that the "mean of log likelihoods" is large. However, as we have mentioned, the above calculations hold only under extremely restrictive conditions, namely, that all the assessors have the same variance, which, on top of this, has to be small ($\sigma^2 < 1/2\pi$). This makes the fact that our algorithm possesses such property of somewhat dubious value as a predictor of good performance in practice.

However, the algorithm has several good (and more important) features:

- it computes both the rankings of items ρ_j as well as the trustworthiness τ_i of the raters *simultaneously*, in a kind of self-adapting manner, in which the weight of a rater depends on his divergence from some form of a "consensus" of the majority of the raters, and this dependence is not based on some discrete classification but is continuous;

- the rankings $\boldsymbol{\rho}$ (and the trustworthiness ranks $\boldsymbol{T}$) depend on each other; just as with the PageRank, they are assigned values jointly in a "global" way; the rank to be assigned to an candidate depends not only on evaluations of such an item by all raters who have evaluated it, but it also depends on ratings of all other candidates, because they influence the trust ranks of all raters. This feature should make this method robust with respect to both outliers and collusive attacks, for as long as the colluders do not overwhelm the "good guys"[2]

We now study the convergence of our iterative algorithm. We return to equation (3.27). Since we had to assume that all $c_j = c_0$, the trust ranks τ_{ij} do not depend on the item evaluated, i.e., do not depend on j, we can write it as

$$\frac{\partial}{\partial \rho_j} g(\boldsymbol{\rho}) = 4 \sum_{i\,:\, i\to j} \tau_i(\boldsymbol{\rho})\, E_{ij} - 4\,\rho_j \sum_{i\,:\, i\to j} \tau_i(\boldsymbol{\rho})$$

[2] As we will see later, one can design systems which can detect and handle even such events under reasonable assumptions.

i.e.,

$$\frac{1}{4\sum_{\,i\,:\,i\to j}\tau_i(\boldsymbol{\rho})}\,\frac{\partial}{\partial\rho_j}g(\boldsymbol{\rho}) = \sum_{i\,:\,i\to j}\frac{\tau_i(\boldsymbol{\rho})}{\sum_{\,l\,:\,l\to j}\tau_l(\boldsymbol{\rho})}\,E_{ij} - \rho_j \tag{3.30}$$

Let, as before, $\boldsymbol{\rho}^{\,(k)}$ denote the "temporary value" of the "consensus" ranks, i.e., the their values after k many iterations of our algorithm. Then, by substituting $\boldsymbol{\rho}$ with $\boldsymbol{\rho}^{\,(k)}$ in equation (3.30) and by noting that the sum on the right is just the new, updated value $\boldsymbol{\rho}^{\,(k+1)}$, we get that for all j, $(1 \leq j \leq M)$,

$$\rho_j^{(k+1)} = \rho_j^{(k)} + \frac{1}{4\sum_{\,i\,:\,i\to j}\tau_i(\boldsymbol{\rho}^{\,(k)})}\,\frac{\partial}{\partial\rho_j}\,g(\boldsymbol{\rho}^{\,(k)}) \tag{3.31}$$

Here we have to make another assumption not mentioned in the paper, namely that the "total trustworthines" of raters rating each item j is similar for all items, so that for all j,

$$\sum_{i\,:\,i\to j}\tau_i(\boldsymbol{\rho}^{\,(k)}) \approx \alpha(k)$$

which in practice is most likely badly violated. With this additional assumption, putting the above equations together to obtain a single vector equation we get

$$\boldsymbol{\rho}^{\,(k+1)} \approx \boldsymbol{\rho}^{\,(k)} + \frac{1}{4\alpha(k)}\,\nabla g(\boldsymbol{\rho}^{\,(k)}) \tag{3.32}$$

which means that our algorithm essentially behaves as the gradient ascent, always updating the values by moving (in our case only approximately) in the direction of the steepest ascent $\nabla g(\boldsymbol{\rho}^{\,(k)})$; see the figure below, which I have "lifted" from another, 2 page paper of the same authors and on the same result, *"Reputation Systems and Optimization"*, SIAM News, Volume 41, Number 2, March 2008.

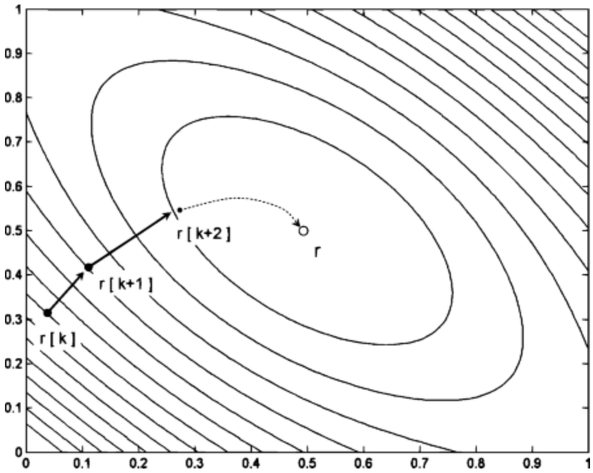

Figure 3.1: Trajectory of a gradient ascent

On the above figure vectors $\boldsymbol{\rho}^{\,(k)}$ are in $\mathbb{R}^2$ for easy visualisation purpose; thus, there are only two candidates to evaluate. The curves are the *level curves* of the candidateive function $g(\boldsymbol{\rho})$. This means that along these curves the value of $g(\boldsymbol{\rho})$ is constant. The gradient is orthogonal to the tangent on such a curve and this is the direction we move in searching for the maximum of $g(\boldsymbol{\rho})$, which is achieved at the point $\boldsymbol{\rho}$ denoted by the little circle. The fact that the coefficients in front of the corresponding partial derivative in equation (3.31) are not necessarily the same impacts the convergence rate, because instead of moving in the direction of the gradient, this direction will be skewed by differences in the factors of the form $1/\left(4\sum_{\,i\,:\,i\to j}\tau_i(\boldsymbol{\rho}^{\,(k)})\right)$; thus, how much will the next iteration deviate from the direction of the fastest ascent depends on how much these multipliers differ for different items j.

We now examine why the authors went into the trouble of using different constants c_j for different items, despite the fact that the above proofs do not go through. Looking at the fixed point equation (3.25),

$$\boldsymbol{\rho} = \left\langle \sum_{i:i\to j}\frac{c_j - \frac{1}{m_i}\sum_{p:i\to p}(E_{ip}-(\boldsymbol{\rho})_p)^2}{\sum_{k:k\to j}\left(c_j - \frac{1}{m_k}\sum_{l:k\to l}(E_{kl}-(\boldsymbol{\rho})_l)^2\right)}E_{ij}\right\rangle_{1\leq j\leq M}$$

we see that, in order to have positive weights

$$w_{ij} = \frac{c_j - \frac{1}{m_i}\sum_{p:i\to p}(E_{ip} - (\boldsymbol{\rho})_p)^2}{\sum_{k:k\to j}\left(c_j - \frac{1}{m_k}\sum_{l:k\to l}(E_{kl} - (\boldsymbol{\rho})_l)^2\right)} \tag{3.33}$$

the costants c_j must be chosen so that the values of $d_i = \frac{1}{m_i}\sum_{p:i\to p}(E_{ip} - (\boldsymbol{\rho})_p)^2$ must be smaller than c_j, i.e., $c_j > d_i(\boldsymbol{\rho})$ for all i, j such that rater i has evaluated item j and for all $\boldsymbol{\rho}$ in a domain D which is sufficiently large to contain

i. the starting point of our iteration $\boldsymbol{\rho}^{(0)}$ which is, as we saw earlier, for each item simply the mean of evaluations of all raters who have rated that item;

ii. the fixed point $\boldsymbol{\rho}$;

iii. all the intermediate points $\boldsymbol{\rho}^{(k)}$ of the iteration.

Note that these values cannot be changed during the iteration process, i.e., they cannot be a function of $\boldsymbol{\rho}$, because this might make the objective function no longer quasi-concave (to be explained later), and thus not necessarily having a unique extremal point.

However, if the same constant c_0 is chosen, then it would have to be pretty large so that $c_0 > d_i(\boldsymbol{\rho})$ for all i and all $\boldsymbol{\rho} \in D$. Thus, a single outlier among the raters might force a large c_0. However, then in the weights

$$w_{ij} = \frac{c_0 - \frac{1}{m_i}\sum_{p:i\to p}(E_{ip} - (\boldsymbol{\rho})_p)^2}{\sum_{k:k\to j}\left(c_0 - \frac{1}{m_k}\sum_{l:k\to l}(E_{kl} - (\boldsymbol{\rho})_l)^2\right)} \tag{3.34}$$

corresponding to items j which were evaluated only by reasonably accurate raters all the expressions of the form $c_0 - \frac{1}{m_i}\sum_{p:i\to p}(E_{ip} - (\boldsymbol{\rho})_p)^2$ in both the denumerator and denominator in (3.34) would be all nearly equal to c_0, and thus the weighted average would be reduced to a simple arithmetic mean. For that reason the authors chose different constants c_j for different items j so that the corresponding c_j has to be larger than the deviations $d_i(\boldsymbol{\rho})$ only of raters i who evaluated item j, thus preserving the "adaptive" feature of the algorithm.[3]

Back to material for all

What happens if there is a collusion attack? So assume that you have 10 sensors deployed and that your adversary managed to compromise 3 of them. The goal of the adversary is to skew the readings, knowing that the aggregator node uses an iterative filtering algorithm to aggregate the values provided by the sensor nodes. If all three compromised sensors keep sending highly skewed values, both iterative filtering algorithms will assign to them low weights and their attempt will fail. Instead, the attackers (presumably each controlling one of the tree compromised nodes) do something much cleverer: First they use their readings $r(i, t_j)$ at time instants t_j to estimate the true values $v(t_j)$ of the parameter monitored by taking the mean of the readings, $m(t_j) = 1/3\sum_{i=1}^{3} r(i, t_j)$. However, instead of reporting their readings, two of the three sensors will report very skewed values $R(i, t_j)$, and the third colluding sensor will report the value $R(3, t_j) = 1/9(7m(t_j) + R(1, t_j) + R(2, t_j))$, which is very close to the mean of the readings of the 7 non compromised sensors and the outlying values provided by the two compromising sensors. Such a value will be very skewed, but it will appear to the aggregating node as the most accurate one, because the aggregator node starts its iterative filtering with the mean of all readings and then proceeds to evaluate the trustworthiness of all sensors.

We have developed a better way to start the iterative filtering process by providing an initial estimate which is much more robust against collusion, as well as an iterative filtering algorithm which significantly outperforms the previous such algorithm. These algorithms are presented in papers available on the webpage and are implemented in the Mathematica file you can find there as well.

Further Reading

- P. Laureti, L. Moret, Y.-C. Zhang and Y.-K. Yu: Information filtering via Iterative Refinement, available at `http://arxiv.org/pdf/physics/0608166v1.pdf`; published in Europhysics Letters (EPL), no. 75, 1006-1012 , 2006.

[3] The same authors have published a revised version of this paper which seem to address the issues we found with this paper, but I have not had time to check it out: Cristobald De Kerchove and Paul Van Dooren: *Iterative filtering in reputation systems*, SIAM Journal of Matrix Analysis and Applications, vol. 31, No. 4, pp. 1812-1834, 2010. It can be a nice final project topic for you to implement it and test it and then write a report on this paper, from both theoretical and empirical perspective.

- C. de Kerchove, P. VanDooren: Iterative filtering for a Dynamical Reputation System, available at `http://perso.uclouvain.be/paul.vandooren/publications/deKerchoveV07.pdf`;

- C. de Kerchove, P. VanDooren: Reputation Systems and Optimization, SIAM News, Volume 41, Number 2, March 2008.

- C. de Kerchove, P. VanDooren: Iterative Filtering in Reputation Systems, SIAM Journal on Matrix Analysis and Applications, Volume 31, Issue 4, January 2010, pp. 1812-1834.

- M. Rezvani, A. Ignjatovic, E. Bertino, and S. Jha. Secure data aggregation technique for wireless sensor networks in the presence of collusion attacks, School of Computer Science and Engineering, UNSW, Tech. Rep. UNSW-CSE-TR-201319, July 2013 IEEE Transactions on Dependable and Secure Computing, 12(1), (2015) 98-110.

- Rezvani M; Ignjatovic A; Bertino E; Jha S, 2016, A collaborative reputation system based on credibility propagation in WSNs, in Proceedings of the International Conference on Parallel and Distributed Systems ICPADS, pp. 1 - 8, http://dx.doi.org/10.1109/ICPADS.2015.9