3D Computer Vision

Radim Šára   Martin Matoušek

Center for Machine Perception
Department of Cybernetics
Faculty of Electrical Engineering
Czech Technical University in Prague

https://cw.fel.cvut.cz/wiki/courses/tdv/start

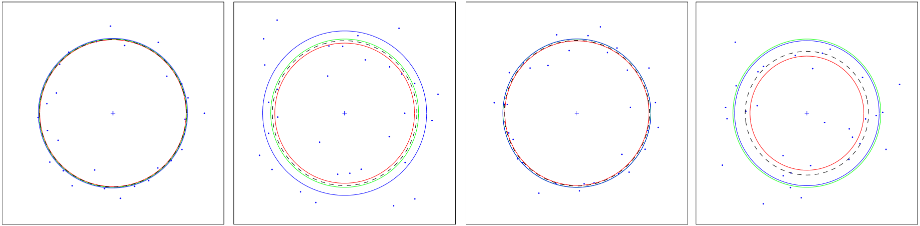
http://cmp.felk.cvut.cz
mailto:sara@cmp.felk.cvut.cz
phone ext. 7203

rev. November 16, 2021

Open Informatics Master's Course

Circle Fitting: Some Results

medium radial noise | big radial noise | medium isotropic noise | big isotropic noise



opt: 1.8, Smp: 1.9, dir: 2.3 | 1.6, 1.8, 2.6 | 1.8, 2.0, 2.2 | 1.6, 2.0, 2.4

mean ranks over 10 000 random trials with $k = 32$ samples

green – ground truth
red – Sampson error $\mathbf{e}$ minimizer
blue – direct radial error $\boldsymbol{\varepsilon}$ minimizer
black – optimal estimator for isotropic error

optimal estimator for isotropic error (black, dashed):

$$r \approx \frac{3}{4k}\sum_{i=1}^{k}\|\mathbf{x}_i\| + \sqrt{\left(\frac{3}{4k}\sum_{i=1}^{k}\|\mathbf{x}_i\|\right)^2 - \frac{1}{2k}\sum_{i=1}^{k}\|\mathbf{x}_i\|^2}$$

which method is better?

- error should model noise, radial noise and isotropic noise behave differently
- ground truth: Normally distributed isotropic error, Gamma-distributed radial error
- Sampson: better for the radial distribution model; Direct: better for the isotropic model
- no matter how corrected, the algebraic error minimizer is not an unbiased parameter estimator

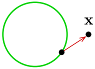
Cramér-Rao bound tells us how close one can get with unbiased estimator and given k

Discussion: On The Art of Probabilistic Model Design…

- a few models for fitting zero-centered circle C of radius r to points in $\mathbb{R}^2$

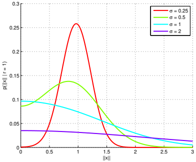
marginalized over C

error model

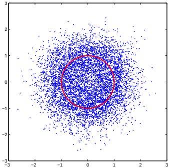


$N(\mathbf{0}, \sigma^2\mathbf{I})$

radial p.d.f.



random sample



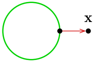
$p(\mathbf{x} \mid r)$

$$\approx \frac{1}{\sigma\sqrt{(2\pi)^3 r\,\|\mathbf{x}\|}}\, e^{-\frac{(\|\mathbf{x}\|-r)^2}{2\sigma^2}}$$

- mode inside the circle
- models the inside well
- tends to normal distrib.

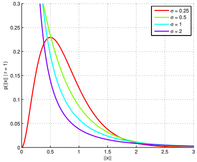
orthogonal deviation from C

error model

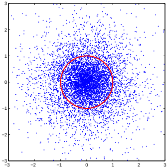


$\Gamma(\cdot, \cdot)$

radial p.d.f.



random sample



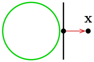
$p(\mathbf{x} \mid r)$

$$\frac{1}{2\pi\Gamma(\frac{r^2}{\sigma})}\,\frac{1}{\|\mathbf{x}\|^2}\left(\frac{r\|\mathbf{x}\|}{\sigma}\right)^{\frac{r^2}{\sigma}} e^{-\frac{r\|\mathbf{x}\|}{\sigma}}$$

- peak at the center
- unusable for small radii
- tends to Dirac distrib.

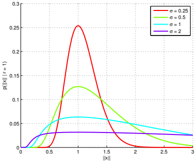
Sampson approximation

error model

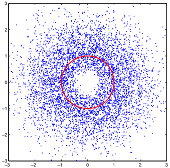


$N(\mathbf{0}, \sigma^2\mathbf{I})$

radial p.d.f.



random sample



$p(\mathbf{x} \mid r)$

$$\frac{1}{r\sigma\sqrt{(2\pi)^3}}\, e^{-\frac{e^2(\mathbf{x};r)}{2\sigma^2}}$$

- mode at the circle
- hole at the center
- tends to normal distrib.

▶Sampson Error for Fundamental Matrix Manifold

The epipolar algebraic error is

$$\varepsilon_i(\mathbf{F}) = \underline{\mathbf{y}}_i^\top \mathbf{F}\, \underline{\mathbf{x}}_i, \quad \mathbf{x}_i = (u_i^1,\, v_i^1), \quad \mathbf{y}_i = (u_i^2,\, v_i^2), \qquad \varepsilon_i \in \mathbb{R}$$

Let $\mathbf{F} = \begin{bmatrix}\mathbf{F}_1 & \mathbf{F}_2 & \mathbf{F}_3\end{bmatrix}$ (per columns) $= \begin{bmatrix}(\mathbf{F}^1)^\top \\ (\mathbf{F}^2)^\top \\ (\mathbf{F}^3)^\top\end{bmatrix}$ (per rows), $\mathbf{S} = \begin{bmatrix}1 & 0 & 0 \\ 0 & 1 & 0\end{bmatrix}$, then

Sampson

$$\mathbf{J}_i(\mathbf{F}) = \left[\frac{\partial \varepsilon_i(\mathbf{F})}{\partial u_i^1}, \frac{\partial \varepsilon_i(\mathbf{F})}{\partial v_i^1}, \frac{\partial \varepsilon_i(\mathbf{F})}{\partial u_i^2}, \frac{\partial \varepsilon_i(\mathbf{F})}{\partial v_i^2}\right] \qquad \mathbf{J}_i \in \mathbb{R}^{1,4}$$

derivatives over point coordinates

$$= \left[(\mathbf{F}_1)^\top \underline{\mathbf{y}}_i,\ (\mathbf{F}_2)^\top \underline{\mathbf{y}}_i,\ (\mathbf{F}^1)^\top \underline{\mathbf{x}}_i,\ (\mathbf{F}^2)^\top \underline{\mathbf{x}}_i\right] = \begin{bmatrix}\mathbf{S}\mathbf{F}^\top \underline{\mathbf{y}}_i \\ \mathbf{S}\mathbf{F}\underline{\mathbf{x}}_i\end{bmatrix}^\top$$

$$\mathbf{e}_i(\mathbf{F}) = -\frac{\mathbf{J}_i(\mathbf{F})\,\varepsilon_i(\mathbf{F})}{\|\mathbf{J}_i(\mathbf{F})\|^2} \qquad \mathbf{e}_i(\mathbf{F}) \in \mathbb{R}^4$$

Sampson error vector

$$e_i(\mathbf{F}) \stackrel{\text{def}}{=} \|\mathbf{e}_i(\mathbf{F})\| = \frac{\varepsilon_i(\mathbf{F})}{\|\mathbf{J}_i(\mathbf{F})\|} = \frac{\underline{\mathbf{y}}_i^\top \mathbf{F}\underline{\mathbf{x}}_i}{\sqrt{\|\mathbf{S}\mathbf{F}\underline{\mathbf{x}}_i\|^2 + \|\mathbf{S}\mathbf{F}^\top \underline{\mathbf{y}}_i\|^2}} \qquad e_i(\mathbf{F}) \in \mathbb{R}$$

scalar Sampson error

- Sampson error 'normalizes' the algebraic error
- automatically copes with multiplicative factors $\mathbf{F} \mapsto \lambda\mathbf{F}$
- actual optimization not yet covered →110

▶Back to Triangulation: The Golden Standard Method

Given $\mathbf{P}_1$, $\mathbf{P}_2$ and a correspondence $x \leftrightarrow y$, look for 3D point $\mathbf{X}$ projecting to x and y →89

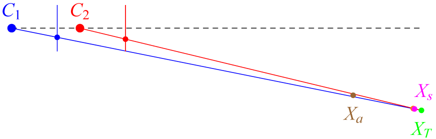
Idea:

1. if not given, compute $\mathbf{F}$ from $\mathbf{P}_1$, $\mathbf{P}_2$, e.g. $\mathbf{F} = (\mathbf{Q}_1\mathbf{Q}_2^{-1})^\top [\mathbf{q}_1 - (\mathbf{Q}_1\mathbf{Q}_2^{-1})\mathbf{q}_2]_\times$
2. correct the measurement by the linear estimate of the correction vector →101

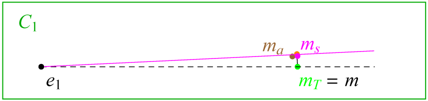
$$\begin{bmatrix} \hat{u}^1 \\ \hat{v}^1 \\ \hat{u}^2 \\ \hat{v}^2 \end{bmatrix} \approx \begin{bmatrix} u^1 \\ v^1 \\ u^2 \\ v^2 \end{bmatrix} - \frac{\varepsilon}{\|\mathbf{J}\|^2}\mathbf{J}^\top = \begin{bmatrix} u^1 \\ v^1 \\ u^2 \\ v^2 \end{bmatrix} - \frac{\underline{\mathbf{y}}^\top \mathbf{F} \underline{\mathbf{x}}}{\|\mathbf{S}\mathbf{F}\underline{\mathbf{x}}\|^2 + \|\mathbf{S}\mathbf{F}^\top\underline{\mathbf{y}}\|^2} \begin{bmatrix} (\mathbf{F}_1)^\top \underline{\mathbf{y}} \\ (\mathbf{F}_2)^\top \underline{\mathbf{y}} \\ (\mathbf{F}^1)^\top \underline{\mathbf{x}} \\ (\mathbf{F}^2)^\top \underline{\mathbf{x}} \end{bmatrix}$$

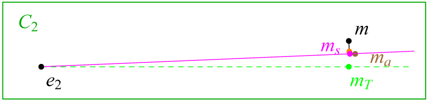
3. use the SVD triangulation algorithm with numerical conditioning →90

Ex (cont'd from →93):



- X_T – noiseless ground truth position
- ● – reprojection error minimizer
- X_s – Sampson-corrected algebraic error minimizer
- X_a – algebraic error minimizer
- m – measurement (m_T with noise in v^2)





▶Back to Fundamental Matrix Estimation

Goal: Given a set $X = \{(x_i, y_i)\}_{i=1}^{k}$ of $k \gg 7$ inlier correspondences, compute a statistically efficient estimate for fundamental matrix $\mathbf{F}$.

What we have so far

- 7-point algorithm for $\mathbf{F}$ (5-point algorithm for $\mathbf{E}$) →84
- definition of Sampson error per correspondence $e_i(\mathbf{F} \mid x_i, y_i)$ →105
- triangulation requiring an optimal $\mathbf{F}$

What we need

- correspondence recognition
- an optimization algorithm for many $(k \gg 7)$ correspondences

$$\mathbf{F}^* = \arg\min_{\mathbf{F}} \sum_{i=1}^{k} e_i^2(\mathbf{F} \mid X)$$

- the 7-point estimate is a good starting point $\mathbf{F}_0$

Levenberg-Marquardt (LM) Iterative Optimization in a Nutshell

Consider error function $\mathbf{e}_i(\boldsymbol{\theta}) = f(\mathbf{x}_i, \mathbf{y}_i, \boldsymbol{\theta}) \in \mathbb{R}^m$, with $\mathbf{x}_i, \mathbf{y}_i$ given, $\boldsymbol{\theta} \in \mathbb{R}^q$ unknown

$\theta = \mathbf{F}$, $q = 9$, $m = 1$ for f.m. estimation

Our goal: $\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{k} \|\mathbf{e}_i(\boldsymbol{\theta})\|^2$

Idea 1 (Gauss-Newton approximation): proceed iteratively for $s = 0, 1, 2, \ldots$

$$\boldsymbol{\theta}^{s+1} := \boldsymbol{\theta}^s + \mathbf{d}_s, \quad \text{where} \quad \mathbf{d}_s = \arg\min_{\mathbf{d}} \sum_{i=1}^{k} \|\mathbf{e}_i(\boldsymbol{\theta}^s + \mathbf{d})\|^2 \tag{19}$$

$$\mathbf{e}_i(\boldsymbol{\theta}^s + \mathbf{d}) \approx \mathbf{e}_i(\boldsymbol{\theta}^s) + \mathbf{L}_i\,\mathbf{d},$$

$$(\mathbf{L}_i)_{jl} = \frac{\partial\big(\mathbf{e}_i(\boldsymbol{\theta})\big)_j}{\partial(\boldsymbol{\theta})_l}, \qquad \mathbf{L}_i \in \mathbb{R}^{m,q}$$

typically a long matrix, $m \ll q$

Then the solution to Problem (19) is a set of 'normal' eqs

$$-\underbrace{\sum_{i=1}^{k} \mathbf{L}_i^\top \mathbf{e}_i(\boldsymbol{\theta}^s)}_{\mathbf{e} \in \mathbb{R}^{q,1}} = \underbrace{\left(\sum_{i=1}^{k} \mathbf{L}_i^\top \mathbf{L}_i\right)}_{\mathbf{L} \in \mathbb{R}^{q,q}} \mathbf{d}_s, \tag{20}$$

- $\mathbf{d}_s$ can be solved for by Gaussian elimination using Choleski decomposition of $\mathbf{L}$

  $\mathbf{L}$ symmetric PSD $\Rightarrow$ use Choleski, almost $2\times$ faster than Gauss-Seidel, see bundle adjustment $\rightarrow$141
- beware of rank defficiency in $\mathbf{L}$ when k is small
- such updates do not lead to stable convergence $\longrightarrow$ ideas of Levenberg and Marquardt

Idea 2 (Levenberg): replace $\sum_i \mathbf{L}_i^\top \mathbf{L}_i$ with $\sum_i \mathbf{L}_i^\top \mathbf{L}_i + \lambda \mathbf{I}$ for some damping factor $\lambda \geq 0$

Idea 3 (Marquardt): replace $\lambda \mathbf{I}$ with $\lambda \sum_i \operatorname{diag}(\mathbf{L}_i^\top \mathbf{L}_i)$ to adapt to local curvature:

$$-\sum_{i=1}^{k} \mathbf{L}_i^\top \mathbf{e}_i(\boldsymbol{\theta}^s) = \left(\sum_{i=1}^{k} \left(\mathbf{L}_i^\top \mathbf{L}_i + \lambda \operatorname{diag}(\mathbf{L}_i^\top \mathbf{L}_i) \right) \right) \mathbf{d}_s$$

Idea 4 (Marquardt): adaptive λ — small $\lambda \to$ Gauss-Newton, large $\lambda \to$ gradient descend

1. choose $\lambda \approx 10^{-3}$ and compute $\mathbf{d}_s$
2. if $\sum_i \|\mathbf{e}_i(\boldsymbol{\theta}^s + \mathbf{d}_s)\|^2 < \sum_i \|\mathbf{e}_i(\boldsymbol{\theta}^s)\|^2$ then accept $\mathbf{d}_s$ and set $\lambda := \lambda/10$, $s := s+1$
3. otherwise set $\lambda := 10\lambda$ and recompute $\mathbf{d}_s$

- sometimes different constants are needed for the 10 and 10^{-3}
- note that $\mathbf{L}_i \in \mathbb{R}^{m,q}$ (long matrix) but each contribution $\mathbf{L}_i^\top \mathbf{L}_i$ is a square singular $q \times q$ matrix (always singular for $k < q$)
- λ helps avoid the consequences of gauge freedom $\to$143
- error can be made robust to outliers $\to$113
- we have approximated the least squares Hessian by ignoring second derivatives of the error function (Gauss-Newton approximation) — See [Triggs et al. 1999, Sec. 4.3]
- modern variants of LM are Trust Region methods

LM with Sampson Error for Fundamental Matrix Estimation

Sampson (derived by linearization over point coordinates u^1, v^1, u^2, v^2)

$$e_i(\mathbf{F}) = \frac{\varepsilon_i}{\|\mathbf{J}_i\|} = \frac{\underline{\mathbf{y}}_i^\top \mathbf{F} \underline{\mathbf{x}}_i}{\sqrt{\|\mathbf{S}\mathbf{F}\underline{\mathbf{x}}_i\|^2 + \|\mathbf{S}\mathbf{F}^\top \underline{\mathbf{y}}_i\|^2}} \quad \text{where} \quad \mathbf{S} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

LM (by linearization over parameters $\mathbf{F}$)

$$\mathbf{L}_i = \frac{\partial e_i(\mathbf{F})}{\partial \mathbf{F}} = \cdots = \frac{1}{2\|\mathbf{J}_i\|} \left[\left(\underline{\mathbf{y}}_i - \frac{2e_i}{\|\mathbf{J}_i\|} \mathbf{S}\mathbf{F}\underline{\mathbf{x}}_i \right) \underline{\mathbf{x}}_i^\top + \underline{\mathbf{y}}_i \left(\underline{\mathbf{x}}_i - \frac{2e_i}{\|\mathbf{J}_i\|} \mathbf{S}\mathbf{F}^\top \underline{\mathbf{y}}_i \right)^\top \right] \tag{21}$$

- $\mathbf{L}_i$ in (21) is a 3×3 matrix, must be reshaped to dimension-9 vector $\mathrm{vec}(\mathbf{L}_i)$ to be used in LM
- $\underline{\mathbf{x}}_i$ and $\underline{\mathbf{y}}_i$ in Sampson error are normalized to unit homogeneous coordinate (21) relies on this
- reinforce $\mathrm{rank}\,\mathbf{F} = 2$ after each LM update to stay on the fundamental matrix manifold and $\|\mathbf{F}\| = 1$ to avoid gauge freedom by SVD →111
- LM linearization could be done by numerical differentiation (we have a small dimension here)

▶Local Optimization for Fundamental Matrix Estimation

Summary so far

- Given a set $X = \{(x_i, y_i)\}_{i=1}^{k}$ of $k \gg 7$ inlier correspondences, compute a statistically efficient estimate for fundamental matrix $\mathbf{F}$.
  1. Find the conditioned (→92) 7-point $\mathbf{F}_0$ (→84) from a suitable 7-tuple
  2. Improve the $\mathbf{F}_0^*$ using the LM optimization (→108–109) and the Sampson error (→110) on all inliers, reinforce rank-2, unit-norm $\mathbf{F}_k^*$ after each LM iteration using SVD

Partial conceptualization

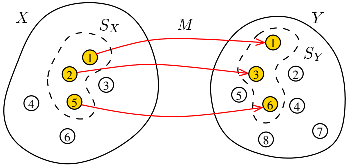
- inlier = correspondence
- outlier = non-correspondence
- binary inlier/outlier labels are hidden
- we can get their likely estimate only, with respect to a model

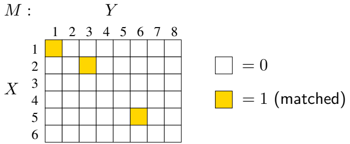
We are not yet done

- if there are no wrong correspondences (mismatches, outliers), this gives a local optimum given the 7-point initial estimate
- the algorithm breaks under contamination of (inlier) correspondences by outliers
- the full problem involves finding the inliers!
- in addition, we need a mechanism for jumping out of local minima (and exploring the space of all fundamental matrices)

Problem: Given image keypoint sets $X = \{x_i\}_{i=1}^{m}$ and $Y = \{y_j\}_{j=1}^{n}$ and their descriptors D, find the most probable

1. inlier keypoints $S_X \subseteq X$, $S_Y \subseteq Y$
2. one-to-one perfect matching $M: S_X \to S_Y$ — perfect matching: 1-factor of the bipartite graph
3. fundamental matrix $\mathbf{F}$ such that $\operatorname{rank} \mathbf{F} = 2$
4. such that for each $x_i \in S_X$ and $y_j = M(x_i)$ it is probable that
   a) the image descriptor $D(x_i)$ is similar to $D(y_j)$, and
   b) the total reprojection error $E = \sum_{ij} e_{ij}^2(\mathbf{F})$ is small — note a slight change in notation: e_{ij}
5. inlier-outlier and outlier-outlier matches are improbable





$$(M^*, \mathbf{F}^*) = \arg\max_{M,\mathbf{F}} p(E, D, \mathbf{F} \mid M)\, P(M) \tag{22}$$

- probabilistic model: an efficient language for problem formulation — it also unifies 4.a and 4.b
- the (22) is a Bayesian probabilistic model — there is a constant number of random variables!
- binary matching table $M_{ij} \in \{0, 1\}$ of fixed size $m \times n$
  - each row/column contains at most one unity
  - zero rows/columns correspond to unmatched point x_i/y_j

Deriving A Robust Matching Model by Approximate Marginalization

For algorithmic efficiency, instead of $(M^*, \mathbf{F}^*) = \arg\max_{M,\mathbf{F}} p(E, D, \mathbf{F} \mid M)\, P(M)$ solve

$$\mathbf{F}^* = \arg\max_{\mathbf{F}} p(E, D, \mathbf{F}) \tag{23}$$

by <u>marginalization</u> of $p(E, D, \mathbf{F} \mid M)\, P(M)$ over $\mathcal{M}$ s.t. $M \in \mathcal{M}$ this changes the problem!

drop the assumption that M is a 1:1 matching, assume correspondence-wise independence:

$$p(E, D, \mathbf{F} \mid M) P(M) = \prod_{i=1}^{m} \prod_{j=1}^{n} p_e(e_{ij}, d_{ij}, \mathbf{F} \mid m_{ij}) P(m_{ij})$$

- e_{ij} represents (reprojection) error for match $x_i \leftrightarrow y_i$: e.g. $e_{ij}(x_i, y_i, \mathbf{F})$
- d_{ij} represents descriptor similarity for match $x_i \leftrightarrow y_i$: e.g. $d_{ij} = \|\mathbf{d}(x_i) - \mathbf{d}(y_j)\|$

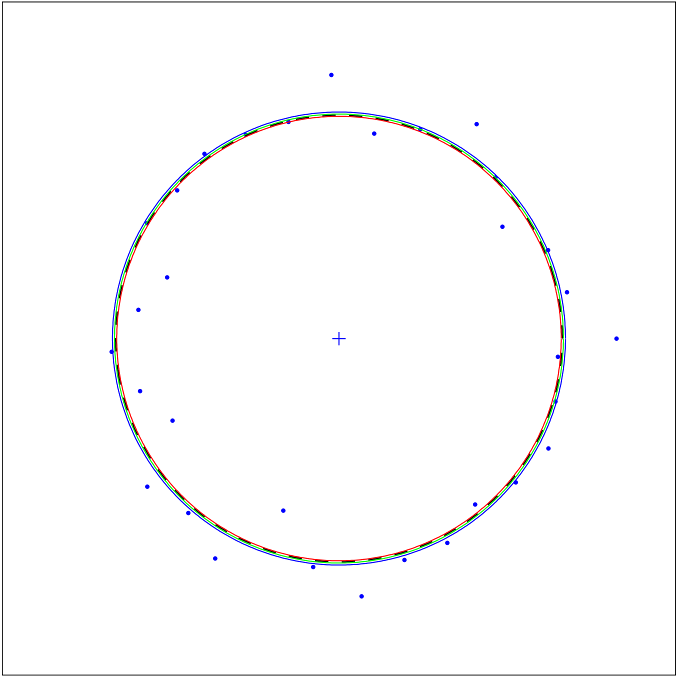
<u>Approximate marginalization:</u> take all the 2^{mn} terms in place of M

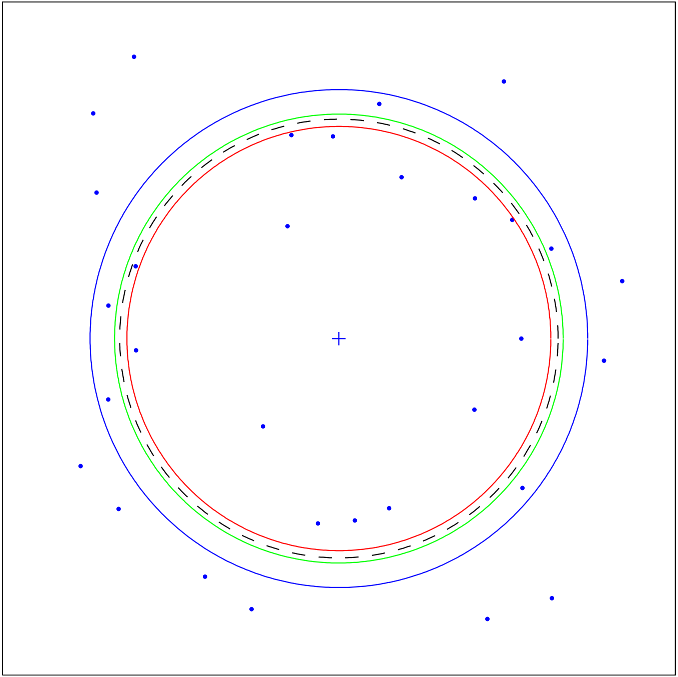
$$p(E, D, \mathbf{F}) \approx \sum_{m_{11} \in \{0,1\}} \sum_{m_{12}} \cdots \sum_{m_{mn}} p(E, D, \mathbf{F} \mid M) P(M) =$$

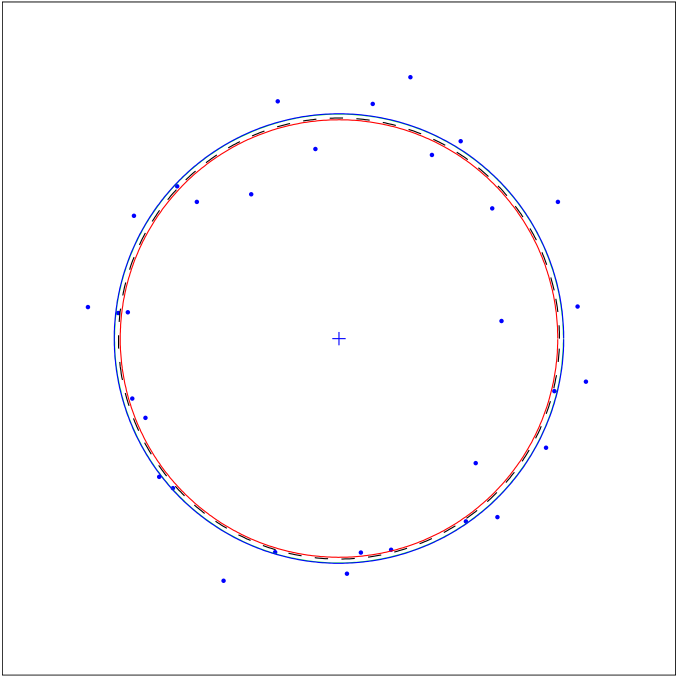
$$= \sum_{m_{11}} \cdots \sum_{m_{mn}} \prod_{i=1}^{m} \prod_{j=1}^{n} p_e(e_{ij}, d_{ij}, \mathbf{F} \mid m_{ij}) P(m_{ij}) \overset{\circledast 1}{=\cdots=}$$

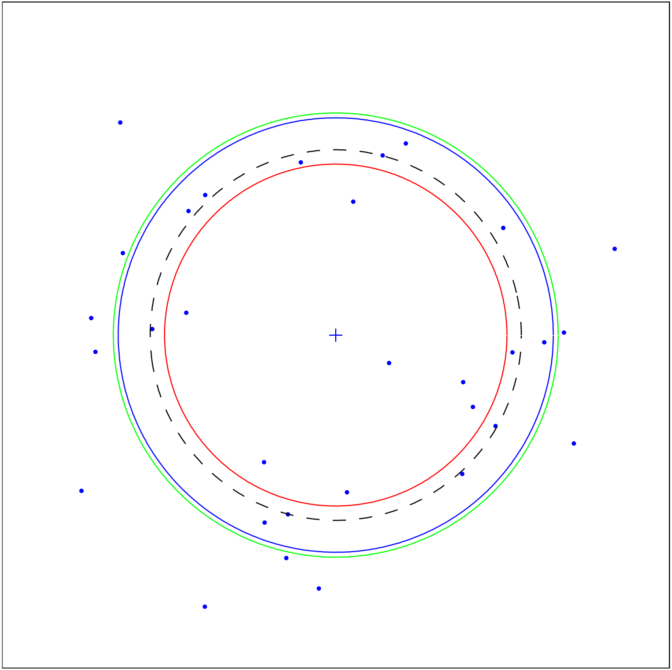
$$= \prod_{i=1}^{m} \prod_{j=1}^{n} \underbrace{\sum_{m_{ij} \in \{0,1\}} p_e(e_{ij}, d_{ij}, \mathbf{F} \mid m_{ij}) P(m_{ij})}_{\text{we will continue with this term}}$$

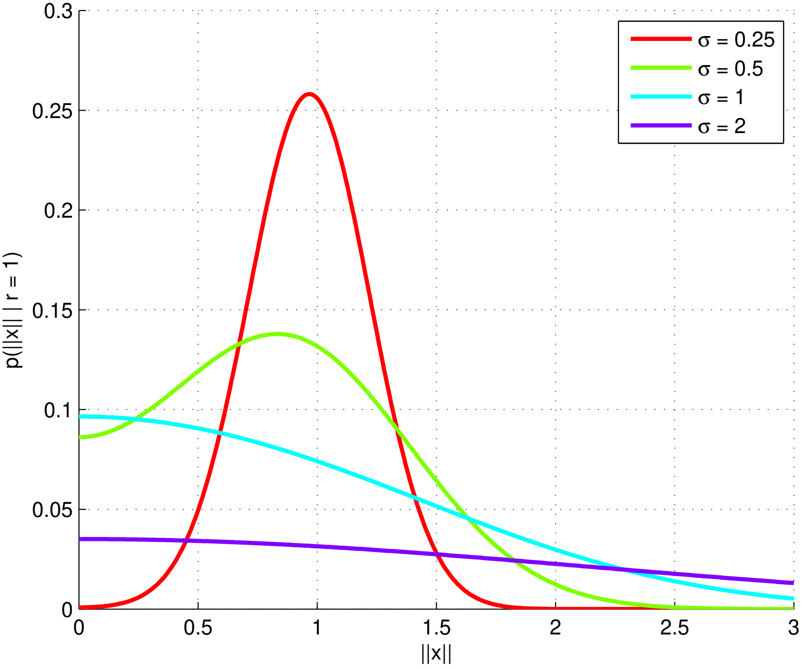
Thank You



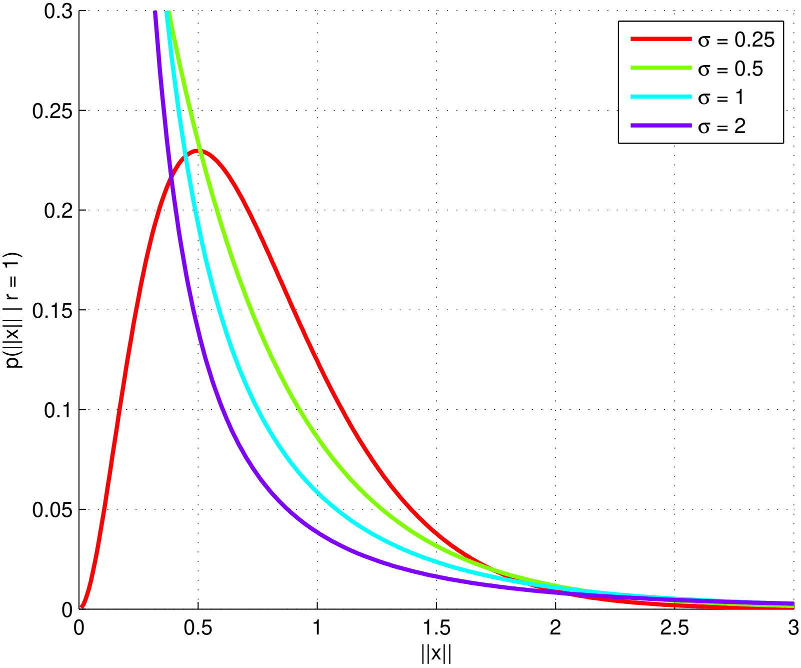


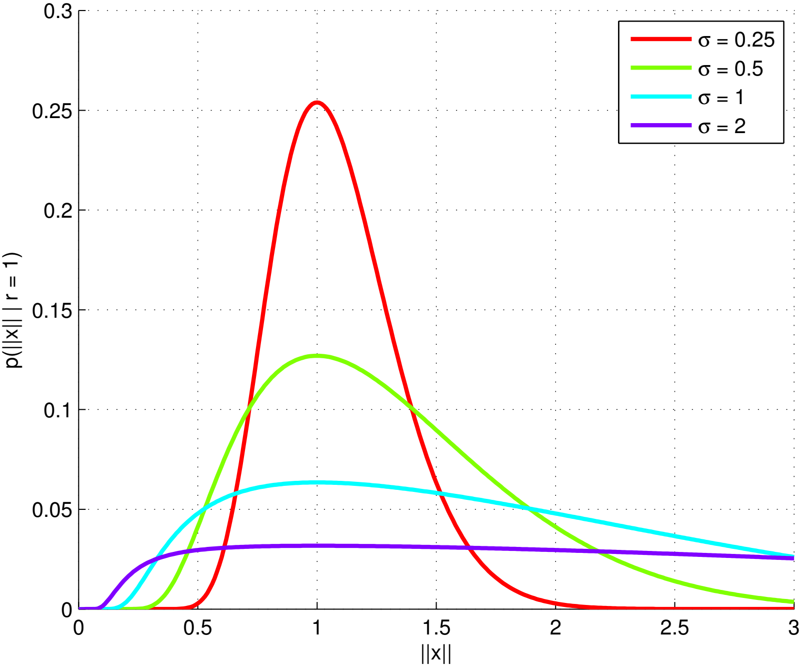


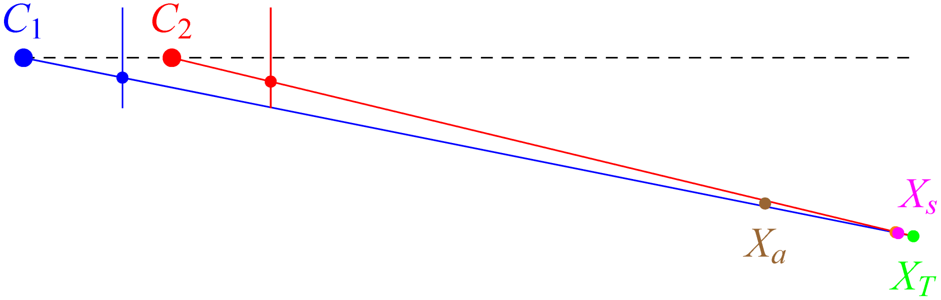




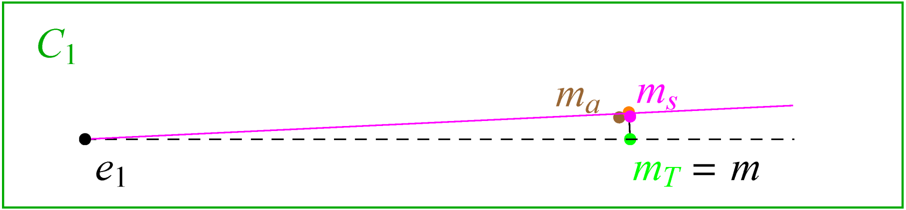
0.3
0.25
0.2
0.15
0.1
0.05
0
p(||x|| | r = 1)
0
0.5
1
1.5
2
2.5
3
||x||
σ = 0.25
σ = 0.5
σ = 1
σ = 2


σ = 0.25
σ = 0.5
σ = 1
σ = 2
p(||x|| | r = 1)
||x||
0.3
0.25
0.2
0.15
0.1
0.05
0
0.5
1
1.5
2
2.5
3

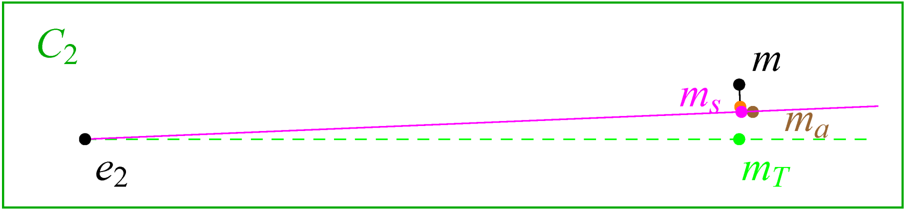

0.3
0.25
0.2
0.15
0.1
0.05
0
p(||x|| | r = 1)
σ = 0.25
σ = 0.5
σ = 1
σ = 2
0
0.5
1
1.5
2
2.5
3
||x||



C_1
C_2
X_s
X_a
X_T



C_1
m_a
m_s
e_1
$m_T = m$



C_2
m
m_s
m_a
e_2
m_T