ASEN 5044, Fall 2018

Statistical Estimation for Dynamical Systems

Lecture 1:
Introduction and Overview

Prof. Nisar Ahmed (Nisar.Ahmed@Colorado.edu)

Tues 08/28/2018

University of Colorado Boulder

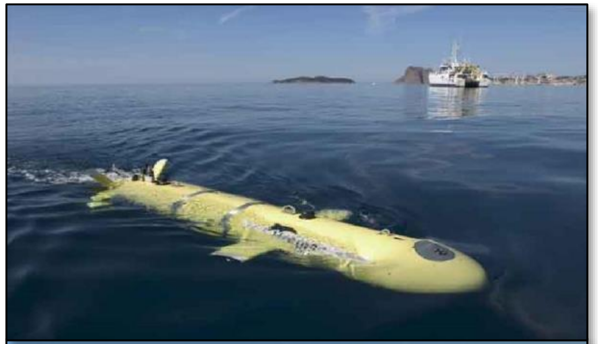
Aerospace Engineering

What's this course about?

- How to tell what's "**really going on**" in a dynamical system?
    - Must know actual behavior to make good **decisions** / do **control** to get desired behavior

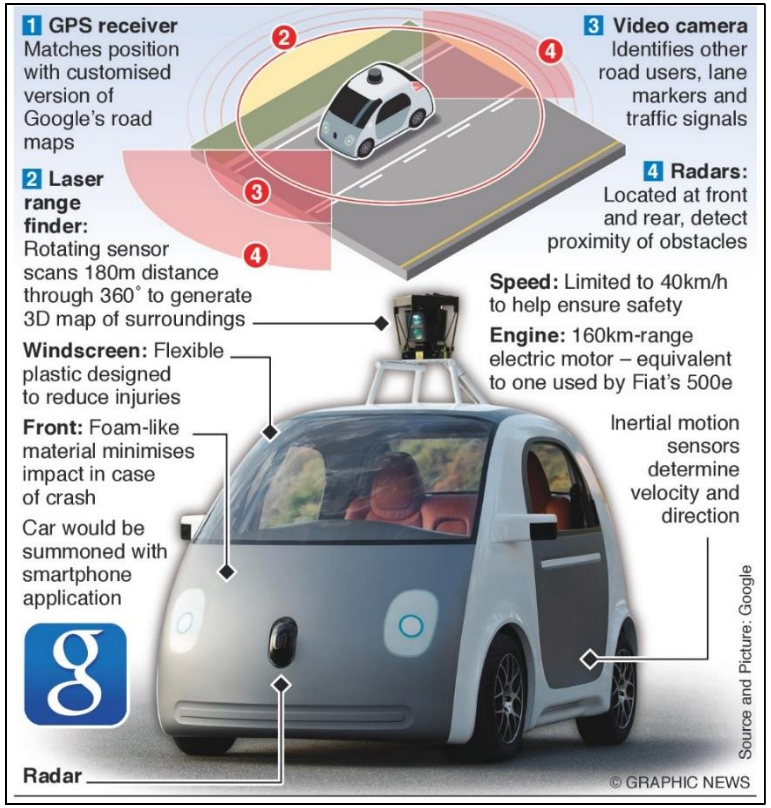
- **Guessing system states** from **noisy data**
    - **Discrete time state space dynamics models** (linear/non-linear systems)
    - Combine with **sensor data** to **invert dynamics** and **infer unknown system variables**

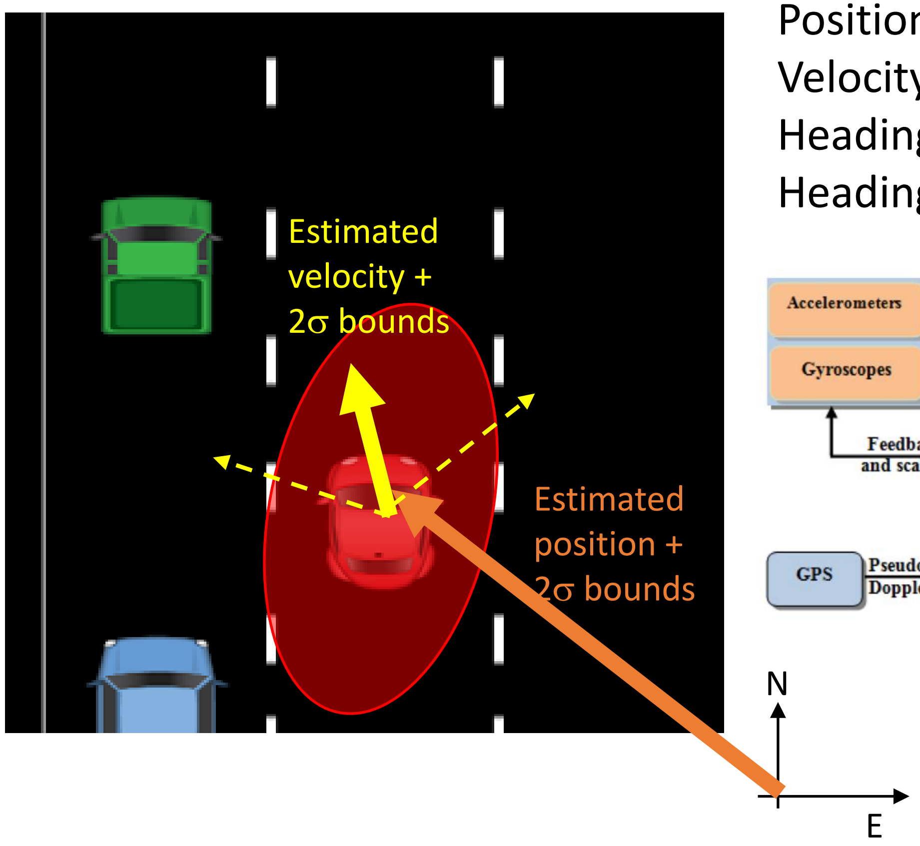
- Combine **engineering models** with **probability and statistics**
    - **Quantitatively describe uncertainty** in **observations** + what **could actually be true**
    - Limits on **observability, accuracy, precision**?
    - How much to **trust** results?



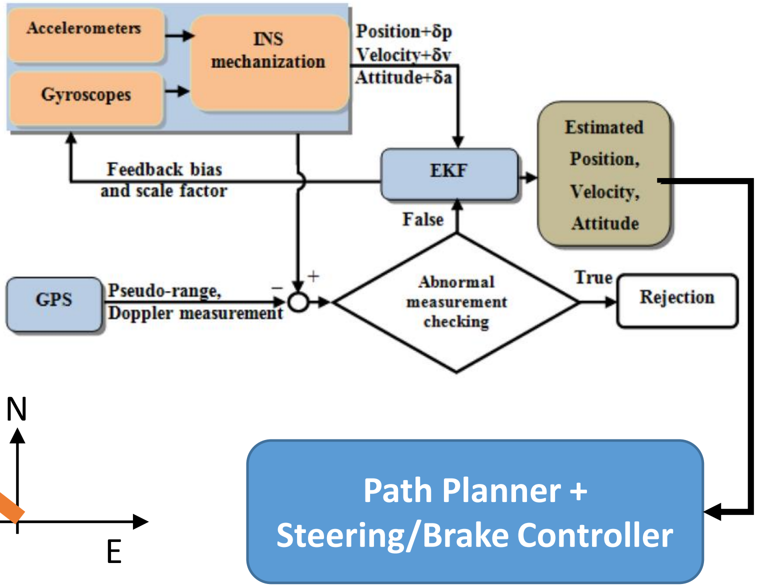
Estimation = Very Educated Guessing

- Example 1: road navigation for a self-driving car
- Most basic question of any vehicle system: where is it and where's it going?



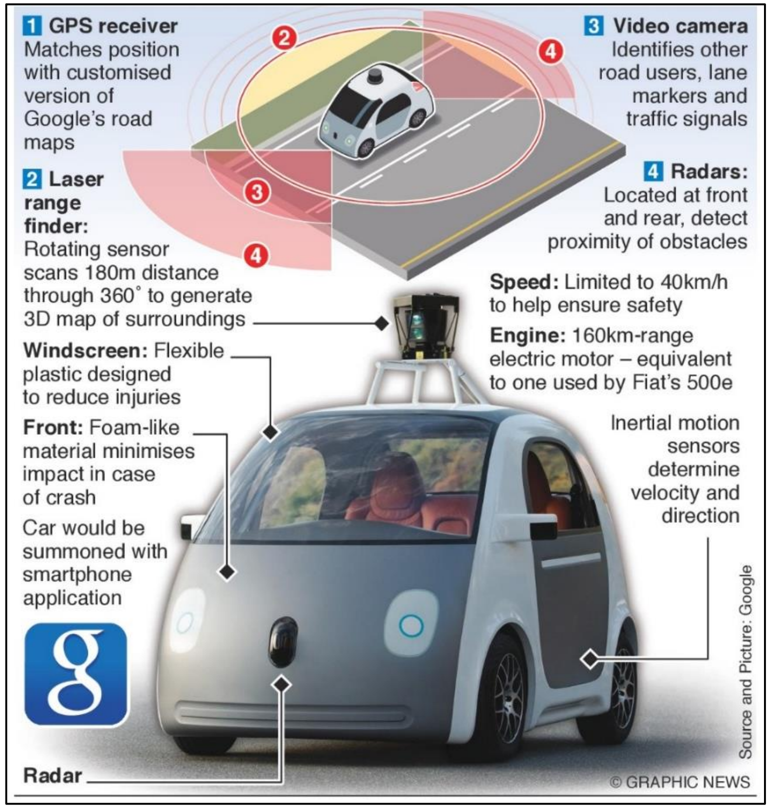


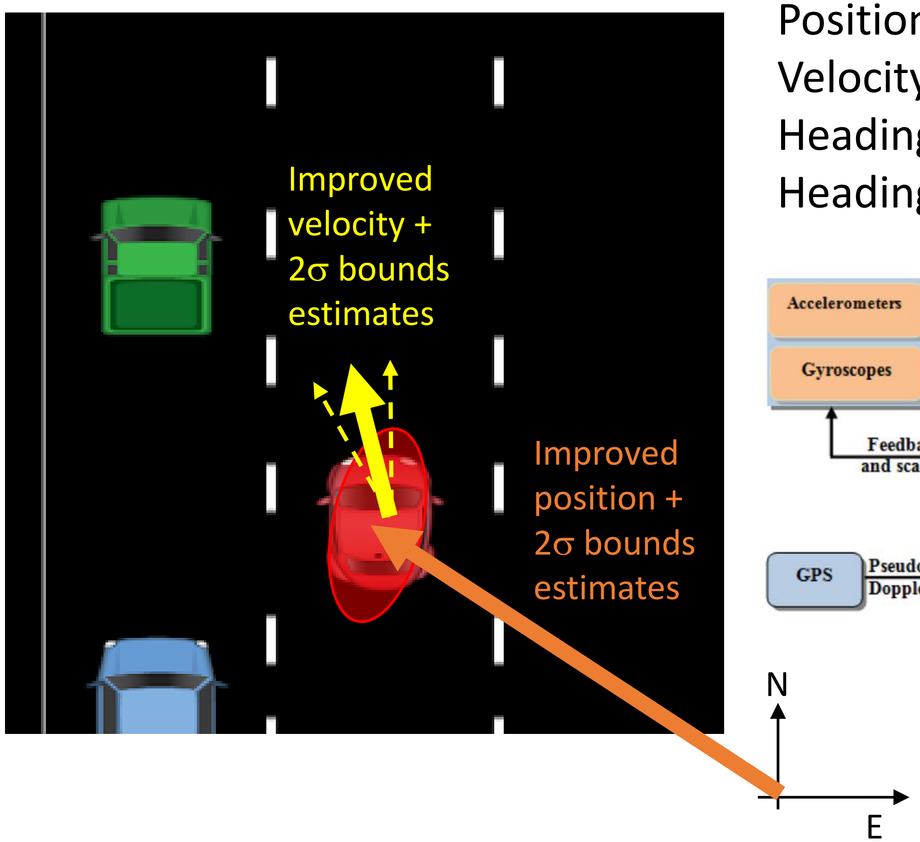
Position =?
Velocity = ?
Heading = ?
Heading rate = ?



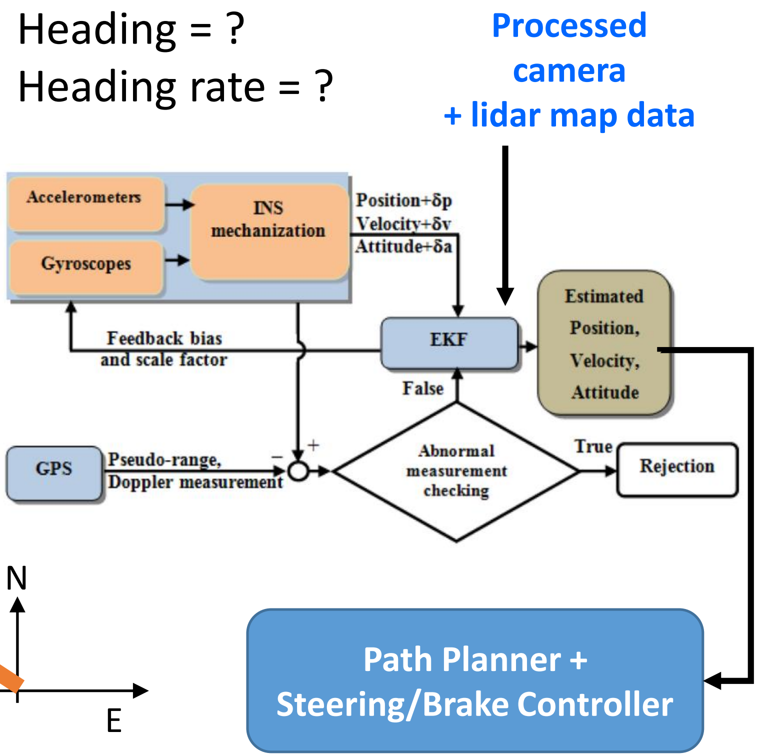
Estimation = Very Educated Guessing

- Example 1: road navigation for a self-driving car
- Most basic question of any vehicle system: where is it and where’s it going?





Position =?
Velocity = ?
Heading = ?
Heading rate = ?

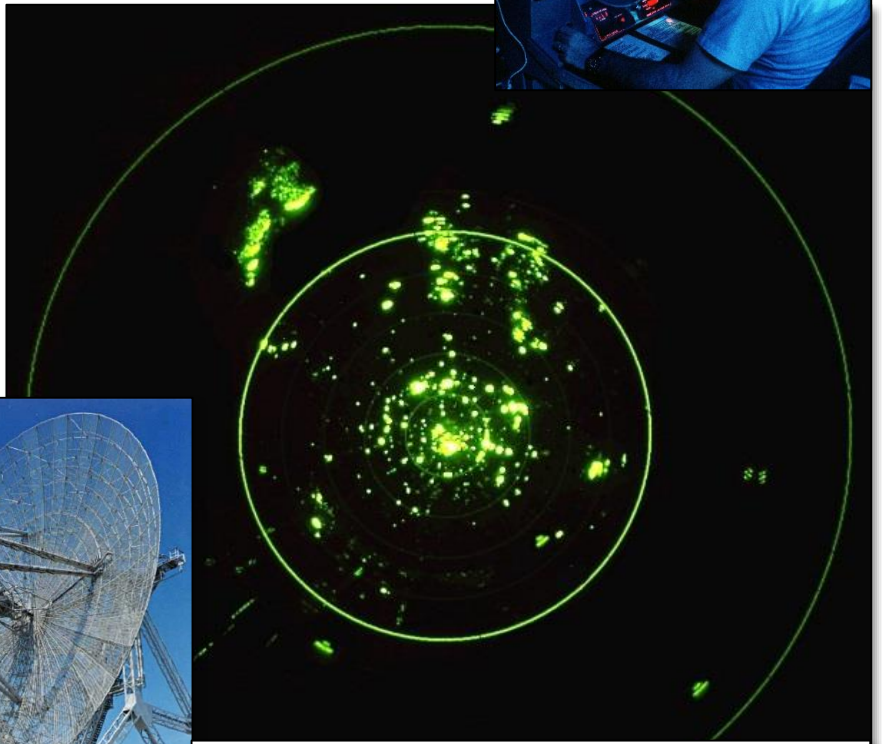


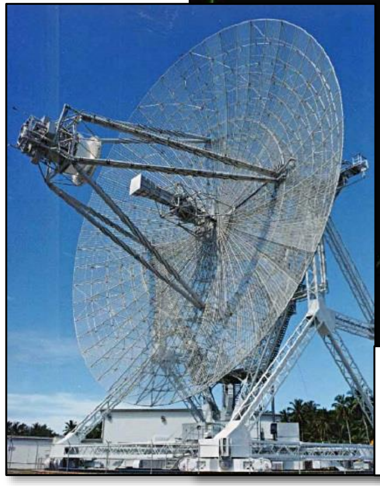
Estimation = Very Educated Guessing

- Example 2: target tracking scenario for air defense

???

???





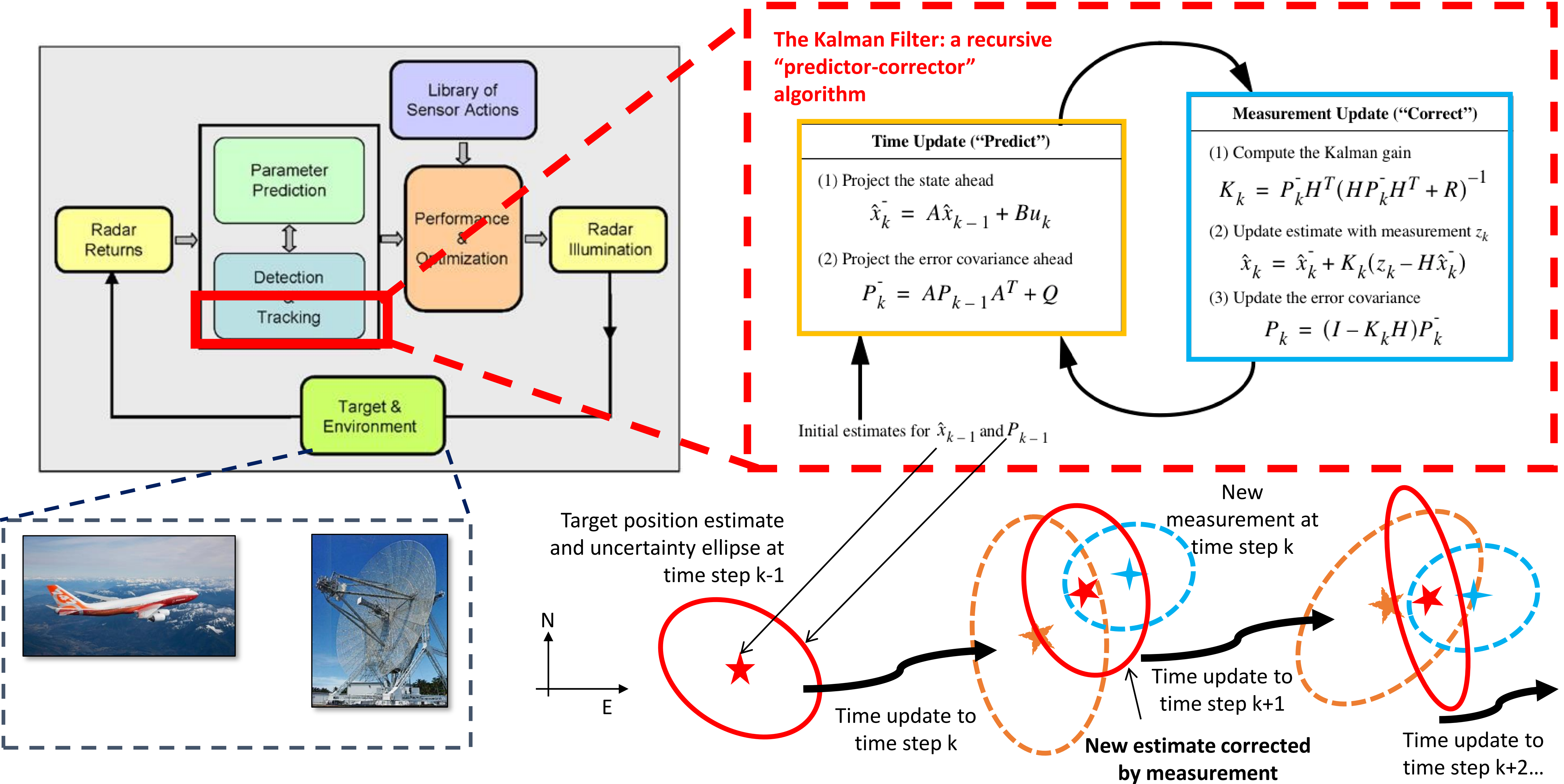
Noisy cluttered radar returns (bearing, range rate, intensity,…)

- Aircraft actually there? How many?
- Position, heading, velocity? Size/shape?
- Friend or foe? Intentions?
- Risk of taking action/no action?

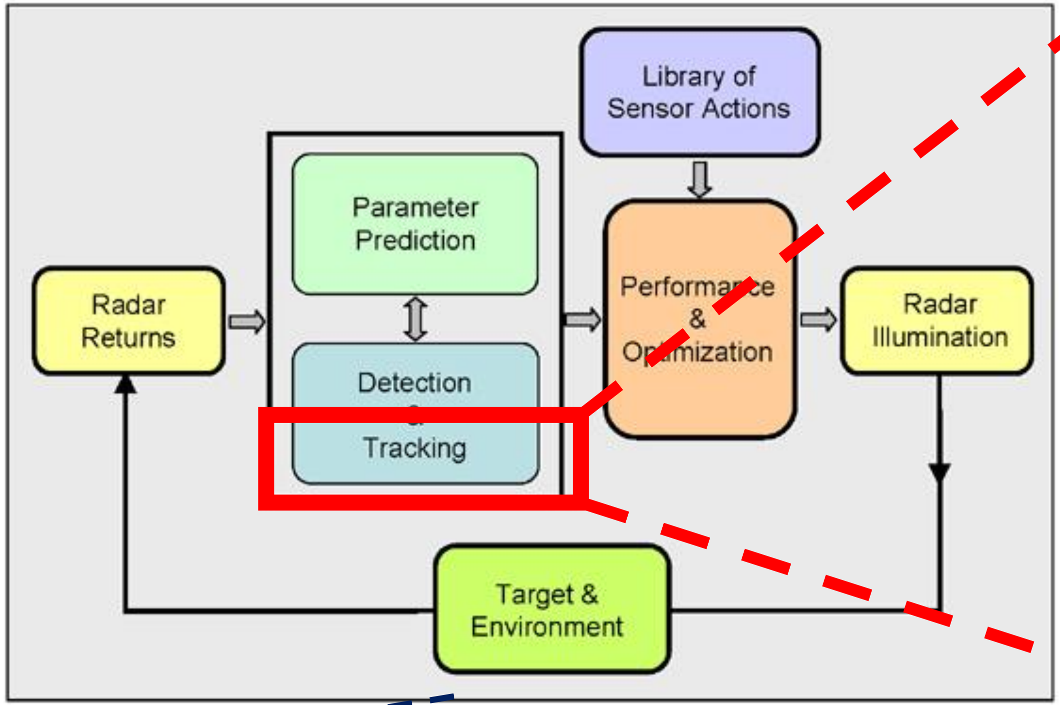
→Combine **models of what we know and don't know** (based on prior engineering experience) with **actual live data** (from noisy sensors)

How to combine? How much data needed? How good do models/sensors have to be? How to get the "best" answers? How much to trust results?

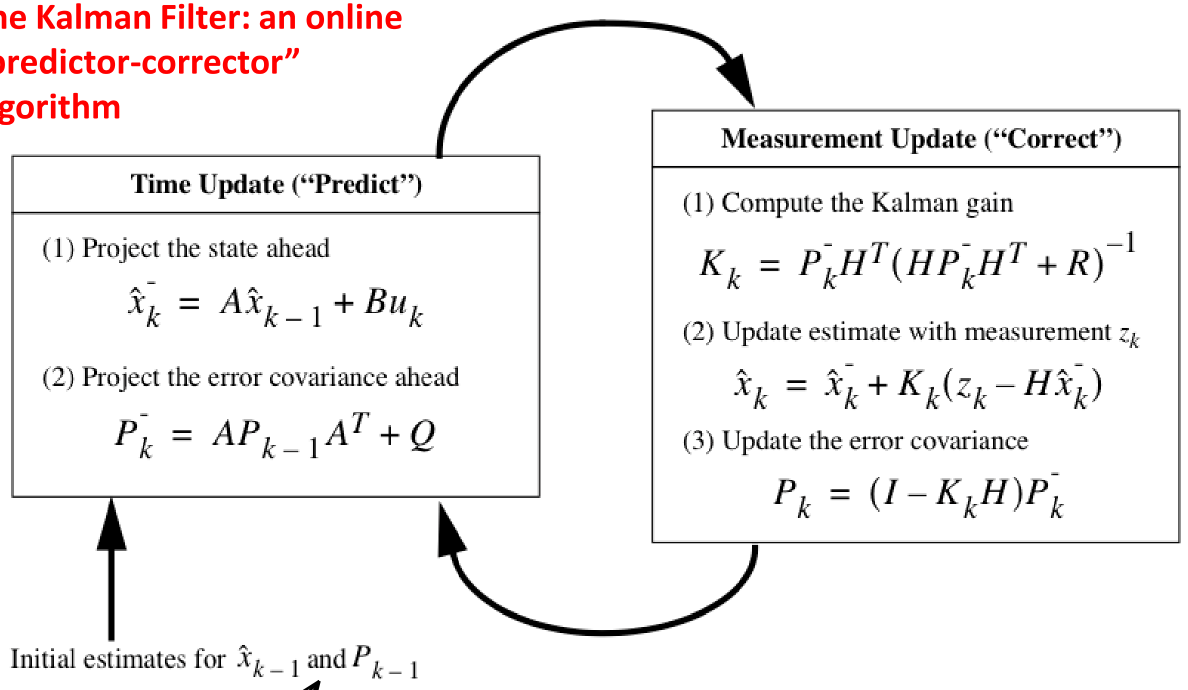
Typical Kalman Filter-Based Radar Target Tracking System

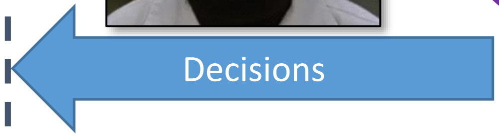


Typical Kalman Filter-Based Radar Target Tracking System

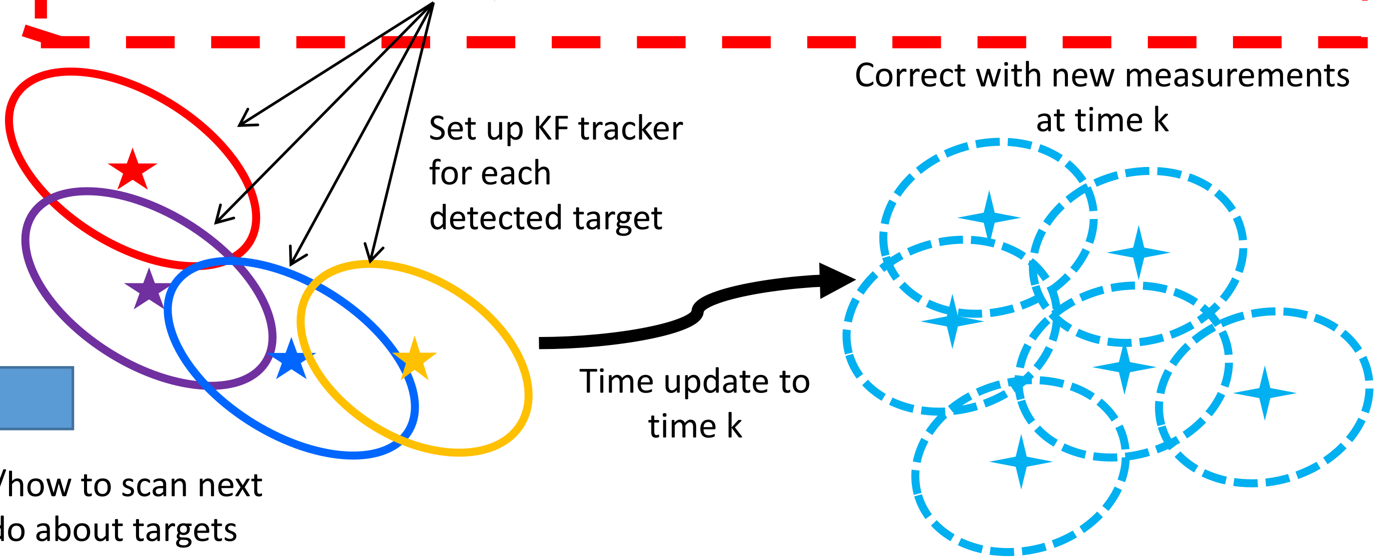


The Kalman Filter: an online "predictor-corrector" algorithm

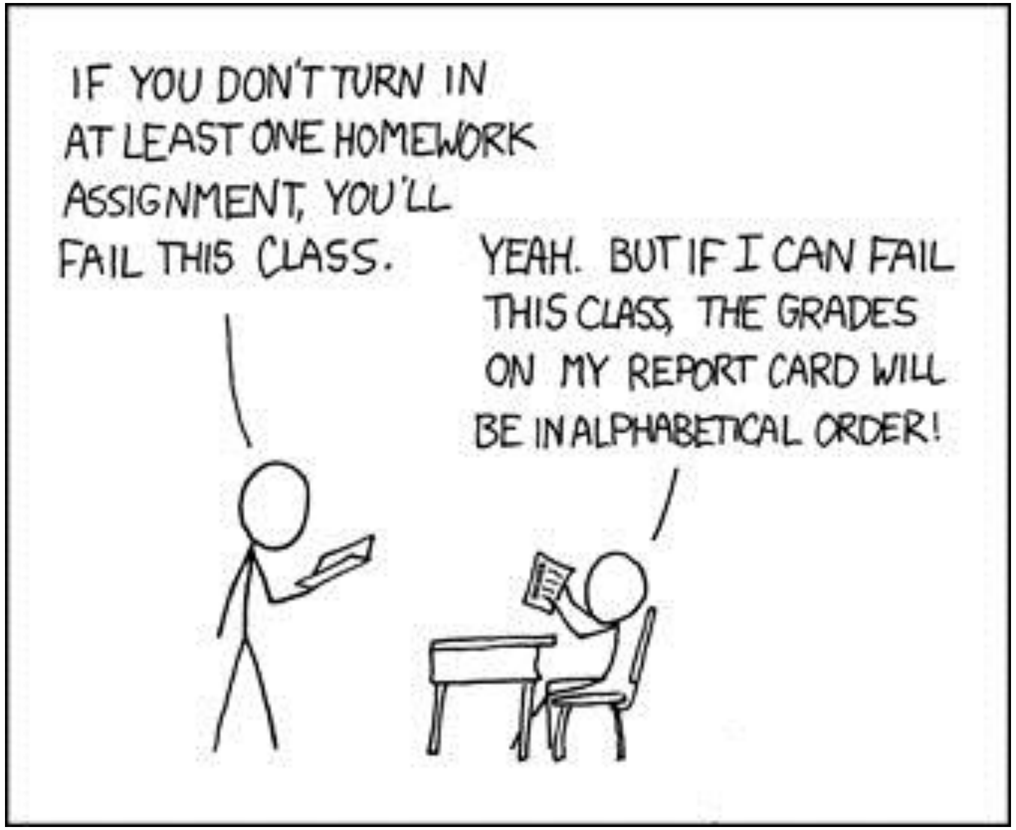




Radar control: where/how to scan next
Response: what to do about targets



Course Overview



Contact Information

Nisar.Ahmed@colorado.edu

Office: ECAE 175

303-492-0286 (office)

Office hours: Tues, 3 pm-4:30 pm

Other appointments: please e-mail me first!

TA Young-young Shen (Aerospace PhD Student)

E-mail: Youngyoung.Shen@colorado.edu

Office Hours: Mon 3-4 pm (Seebass Conference Room, ECAE 153)

Learning Goals

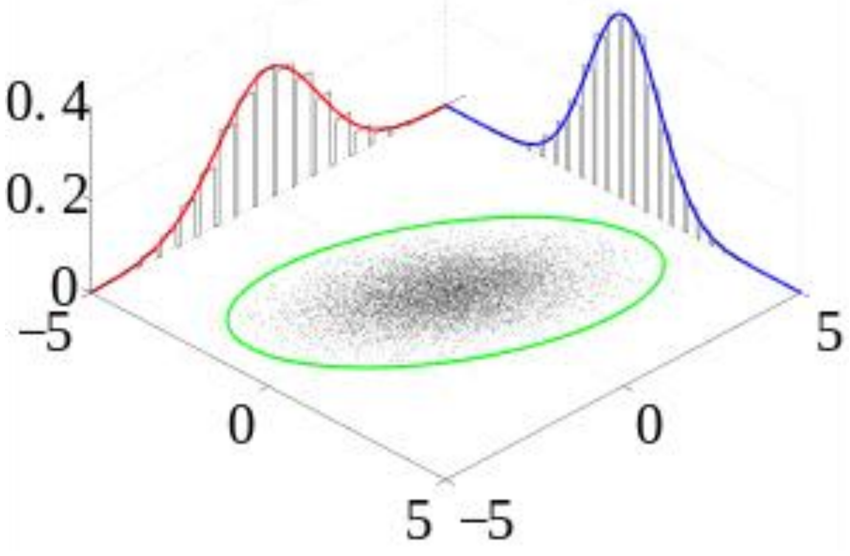
1. Absorb basic theory and engineering usage of probability and statistics

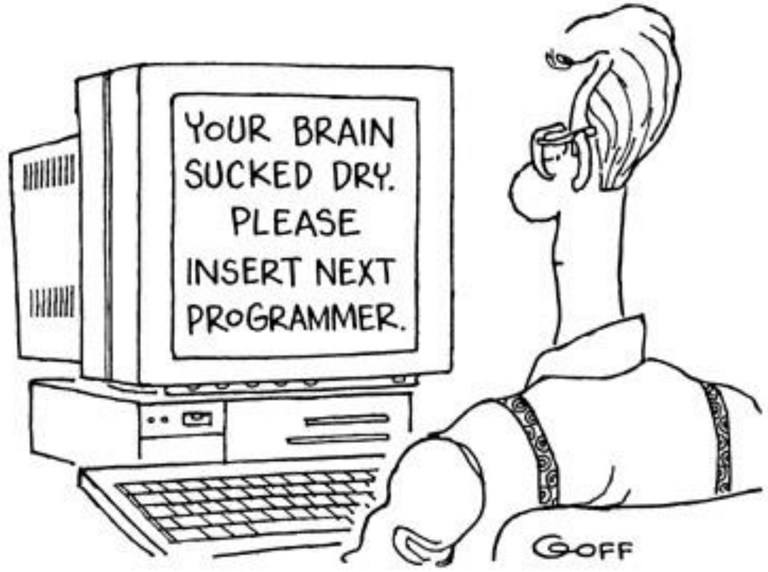
2. Explore, explain and apply core concepts of statistical estimation theory,

    especially to problems defined by (discrete time) stochastic state space models

3. Formulate and solve dynamic state estimation problems using **batch/recursive least squares, Kalman filters**, and other related estimation algorithms

4. Design, **program**, simulate, evaluate, visualize and tune estimator performance on real system models using software tools (e.g. Matlab, Python)





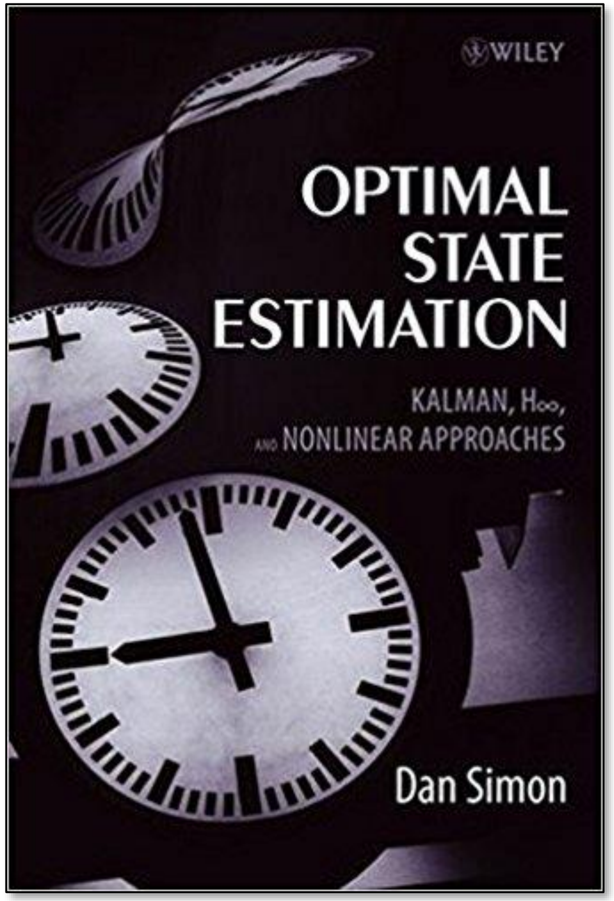
Assessments and Lectures

- **2 midterm exams** (25% each)
- **1 final project** (30%) – must work in pairs, more details later in course
- **Weekly homework** (20%)
  - Assigned on Thurs, due following week on Thurs
  - Partially graded
  - Solutions to problems will be posted
  - Graded/ungraded problems may show up on exams in altered form
  - "Advanced Questions"
    - **required for PhD students to answer** (in addition to regular HW questions)
    - Non-PhD students can answer for extra credit (if regular HW complete and on time)

- Equal emphasis on theory and computation (Matlab, or Python,…)

Textbook and Website

- Dan Simon, *Optimal State Estimation*, Wiley and Sons, 1st Ed., 2006.
  - errata for text available online (see link in syllabus)
  - cheaper e-book version available (wiley.com)
- Course Website: http://canvas.colorado.edu (ASEN 5044, Fall 2018)
  - Assignments
  - Exams
  - Solutions
  - Recorded lectures
  - Announcements
  - Errata in notes/assignments
  - Other Posted Materials
- Supplementary books (completely optional):
  - Crassidis and Junkins, "Optimal Estimation for Dynamical Systems", 2nd ed (available through CU Library as e-book)
  - R. Stengel, "Optimal Control and Estimation," Dover, 1994 **($30!)**



Anticipated Course Schedule and Book Coverage

- Almost all topics are in the Simon textbook
  - Lectures will cover some topics more deeply than book
  - Not all book chapters to be covered in lecture
  - A few topics not in book (e.g. chi-square tests for KF tuning…)
  - Many homework questions will come from textbook
  - **Be sure to read the notes and textbook!**

Week(s)	Topic	Text Chaps.
1	Intro & overview	–
1-3	Basic linear dynamical systems theory, discrete time systems	1.1-1.7
3-6	Basic probability and stochastic process theory	2.1-2.7
6-8	Least squares estimation, stochastic linear systems	3.1-3.7, 4.1-4.2
8-11	The Kalman filter (KF): basics, tuning, testing, generalizations	5.1-5.5, 6, 7
11-14	Nonlinear filters: Linearized KF and EKF	13.1,13.2
14-15	Advanced topics / Guest lectures	8, $\cdots$

  - Special topic extra lectures (weeks 5-11): Bayesian estimation theory
    - Strongly recommended for Ph.D. students, others welcome to attend

Grading

- **Clearly understand the concepts:**
  - correct, precise terminology
  - appropriate method or strategy
  - appropriate tools, built on previous results
  - correct demonstrations (“show”, “prove”) – be as rigorous as possible!

- Legible, careful development:
  - **explain strategy – always show work (laziness/sloppiness will cost you!)**
  - correct calculations -- **always check your numbers!**
  - **use consistent + correct notation**, don’t write too small/turn in illegible work
  - **label plots, use units, draw helpful pictures**
  - cite previous results used (don’t re-invent the wheel)

Grading Scale

- > 90% Clear comprehension, correct results, only minor errors (e.g. sign) → **A**
- ~80-90% Seems to understand, few significant errors → **B+ to A-**
- ~70-80% Major errors, hard to tell if concepts understood → **B**
- ~60-70% Conceptual grasp clearly lacking → **B-**
- < 60% Little understanding is evident → **C**

Grading Scale

No competition for grades:

I hope everyone earns an A!

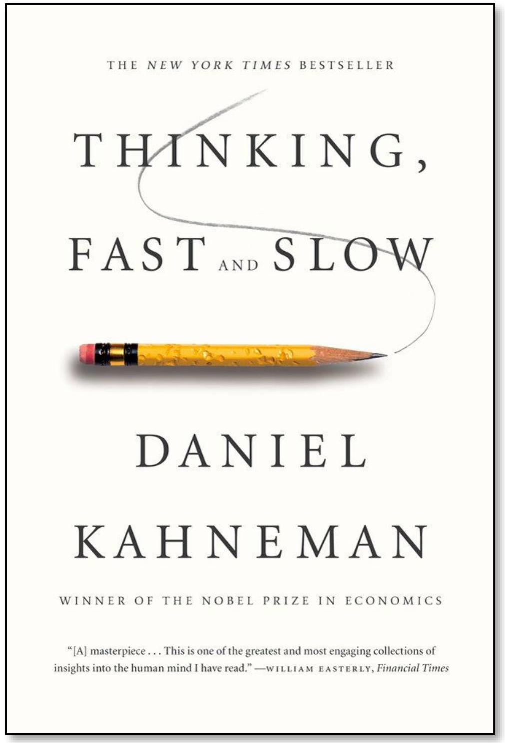
(especially this guy)

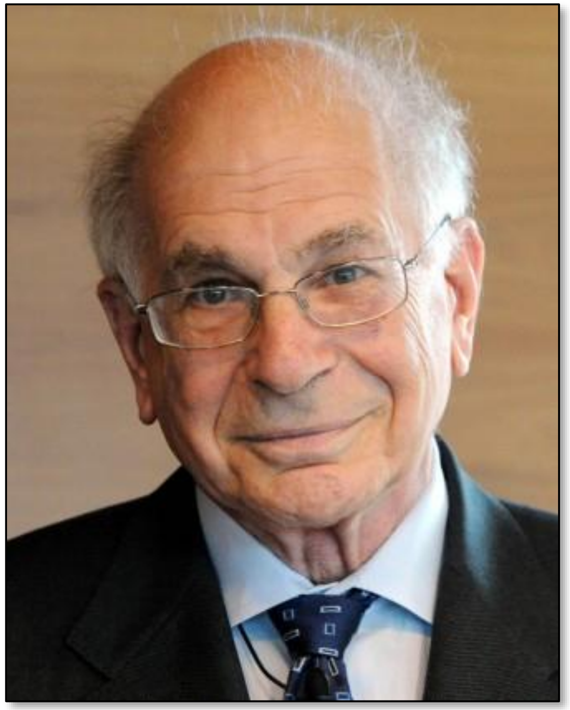
General Recommendations for Grad Classes

- **Learn by doing (both theory and algorithms)**, not by watching
  - Lectures = guide – do not rely on osmosis
- **Read book, take your own notes, read your notes, ask questions!**
- **Spend time** on HW and looking at solutions
- **Watch lectures again** if you didn't write/catch something first time
- OK to do HW in group, **but always try HW on your own first!**
- **Come to office hours,** even if you don't "have to"
- **Be prepared**: study the notes and the book *before* class and office hours
- **Always ask questions** before, during, and after class (email is good too!)
- Start homework, exams and projects **early** – **don't wait until last minute!**
- Always do **proper sanity checks** for problems!
  - **Always try proving things yourself!** (rigor expected -- no hand waving!)
- **Learn to slow down, think out loud, and argue with yourself**

Thinking Fast and Slow: Your Brain on Stats

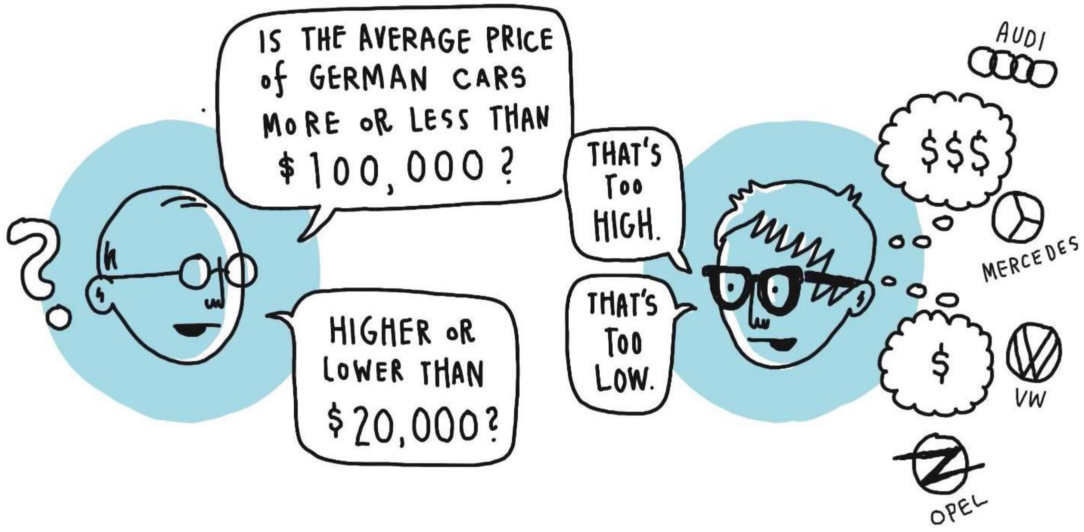
- Our brains are big messy estimators, **which we must learn to tame**…





Nobel Laureate Behavioral Economist and Psychologist

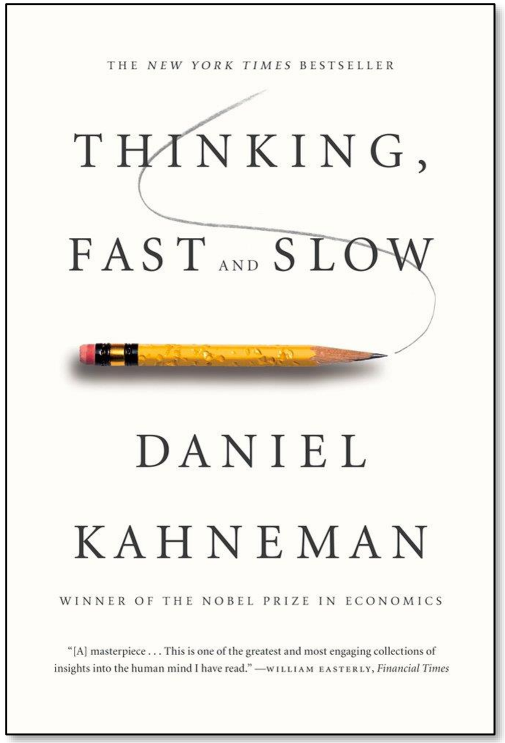
The availability heuristic

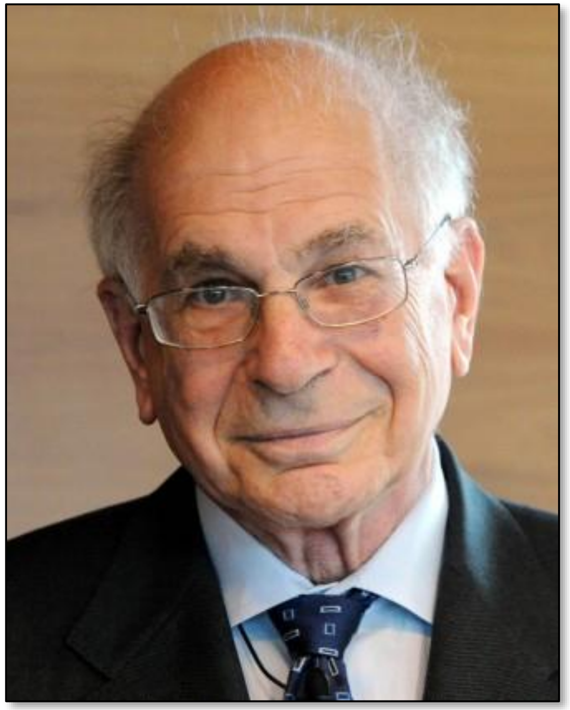


https://i.ytimg.com/vi/HefjkqKCVpo/maxresdefault.jpg

Thinking Fast and Slow: Your Brain on Stats

- Our brains are big messy estimators, **which we must learn to tame…**





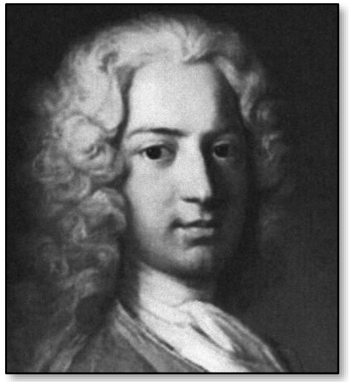
Nobel Laureate Behavioral Economist and Psychologist

The anchoring effect

The framing effect

FRAMING	*TREATMENT A*	*TREATMENT B*
Positive	"***Saves*** 200 Lives"	"A 33% chance of ***saving*** all 600 people, 66% possibility of ***saving*** no one"
Negative	"400 People Will ***Die***"	"A 33% chance that no people will ***die***, 66% probability that all 600 people will ***die***"

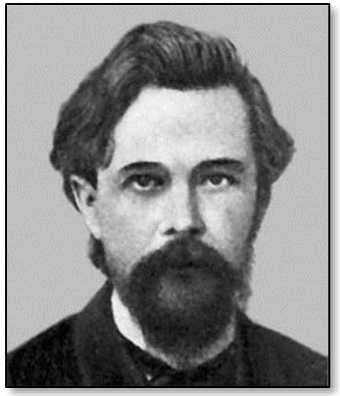

T. Bayes **(??)**


D. Bernoulli

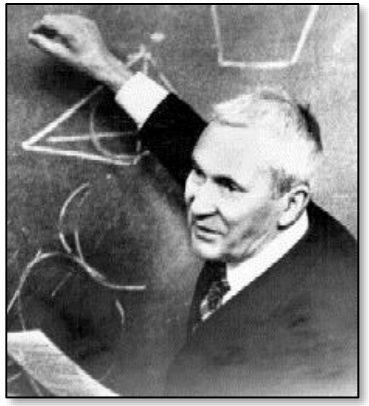

A-M. Legendre

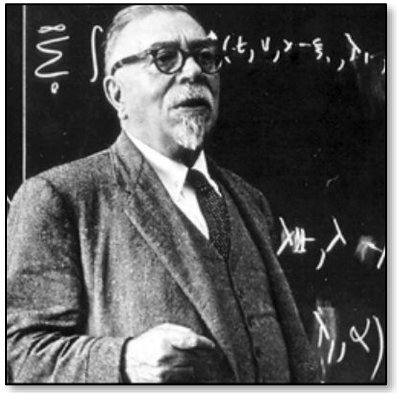

K. Gauss

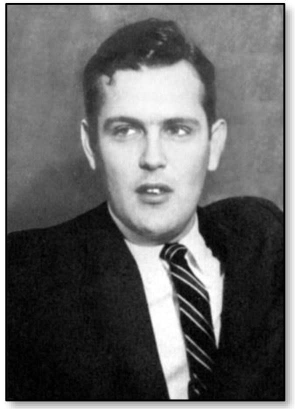

P-S. Laplace

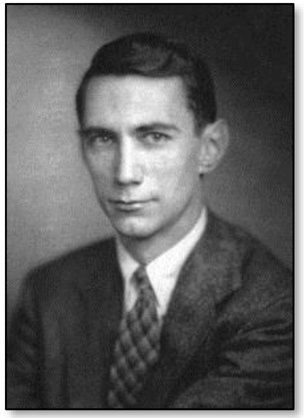

A. Markov

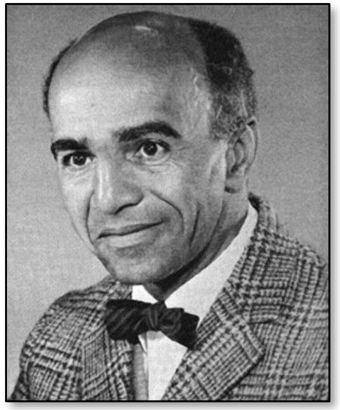

R. Fisher


A. Kolmogorov


N. Wiener

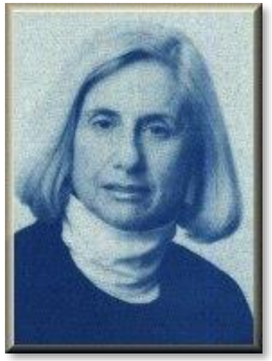

R. Kalman


C. Shannon


D. Blackwell

Onto Estimation: The Artful Science of Guessing in a (very small and incomplete) Nutshell


C.R. Rao


G. Wahba

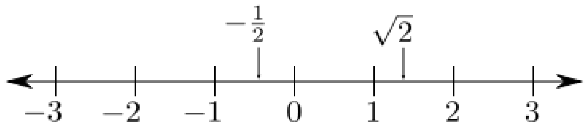
Estimation = Very Educated Guessing

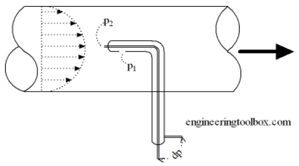
- Estimation = process of inferring a quantity of interest from indirect, inaccurate, and uncertain observations

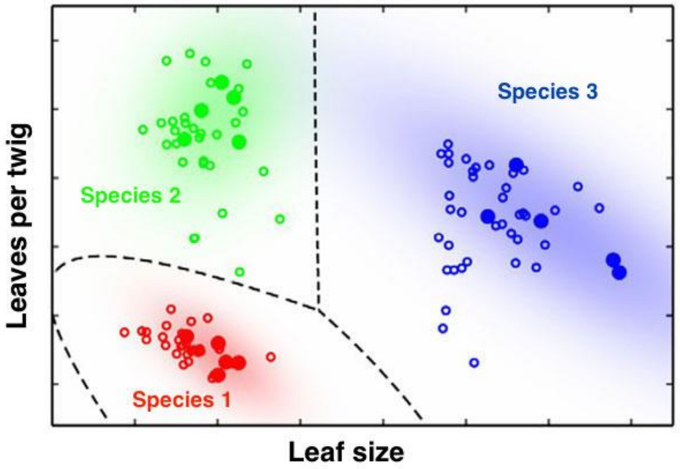
- More rigorous definitions: two major flavors [need both in practice]
  - **continuous estimation**: select “best” point from an infinite continuous space
  - **discrete decision-making**: choosing from a finite set of exclusive alternatives (hypothesis testing, classification, data association, model selection, …)

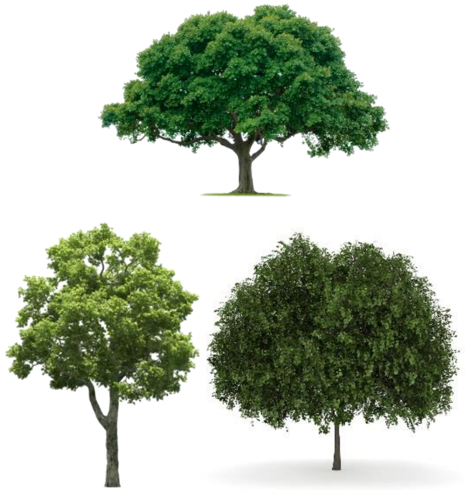












Estimation = Very Educated Guessing

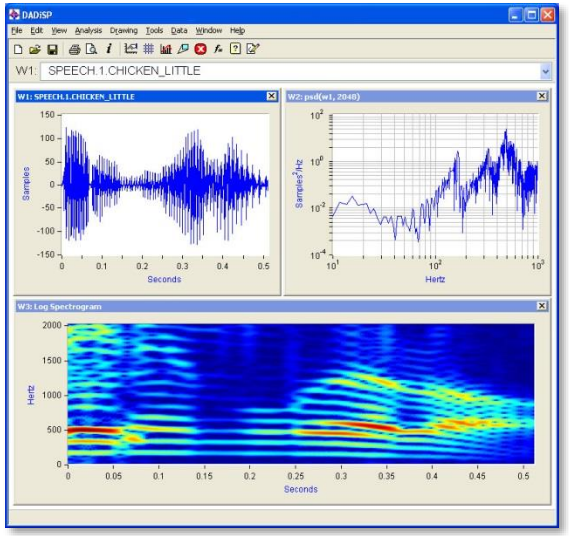
- Generally: *information extraction and enhancement*

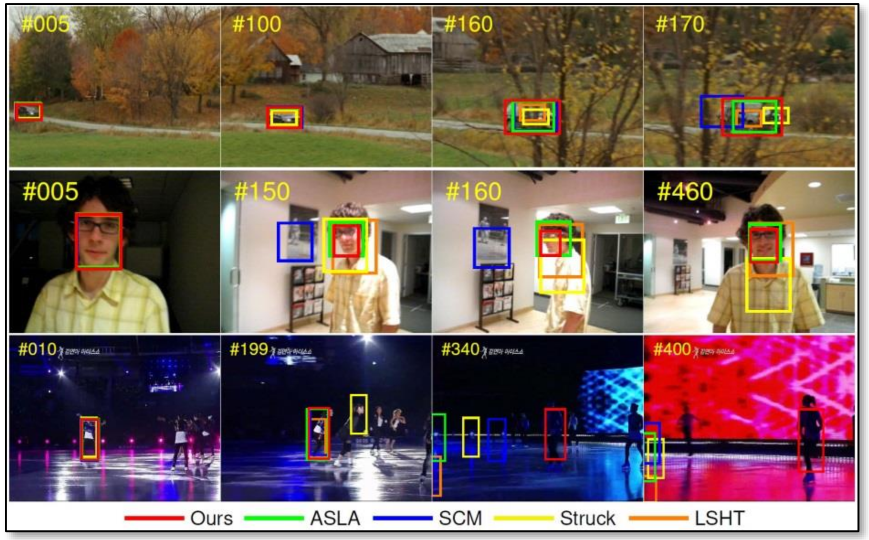
- Major types of estimation problems
  - **Parameter estimation:** determine the values of a model that govern system behavior
  - **Filtering:** “clean up” noisy signals over time to find **current system state**
  - **Prediction:** calculate evolution of **future system states** given current information
  - **Smoothing:** use info available now to improve knowledge of **past system states**
  - **Batch processing:** combine all info over a time window to estimate all system states at once

- In this course, we focus on **model-based estimation problems**

Estimation = Very Educated Guessing

- Common engineering applications of model-based estimation
  - **Tracking:** find state of moving object using *remote* sensor measurements
  - **Navigation:** find state of platform *where sensors located* (car, UAV, boat, iPhone,…)
  - **Pattern recognition:** build models to predict labels of new data based on old training data
  - **System identification:** determine governing equations of dynamical system from data
  - **Fault monitoring/diagnosis:** determine whether system operating in normal/nominal state



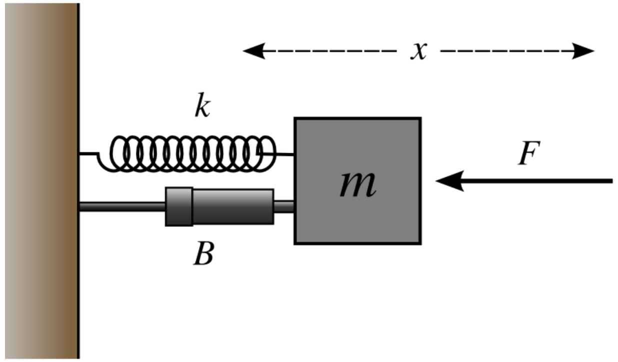


Parameter Estimation

- General problem statement: given some function/model f(k,x) = y and series of observations y_i, i = 1,…N, how to find parameters (k, B)?
  - if x_i known/unknown?

- Control engineering examples (system ID):
  - linear model: find (A,B,C,D) matrices from (u,y) data:

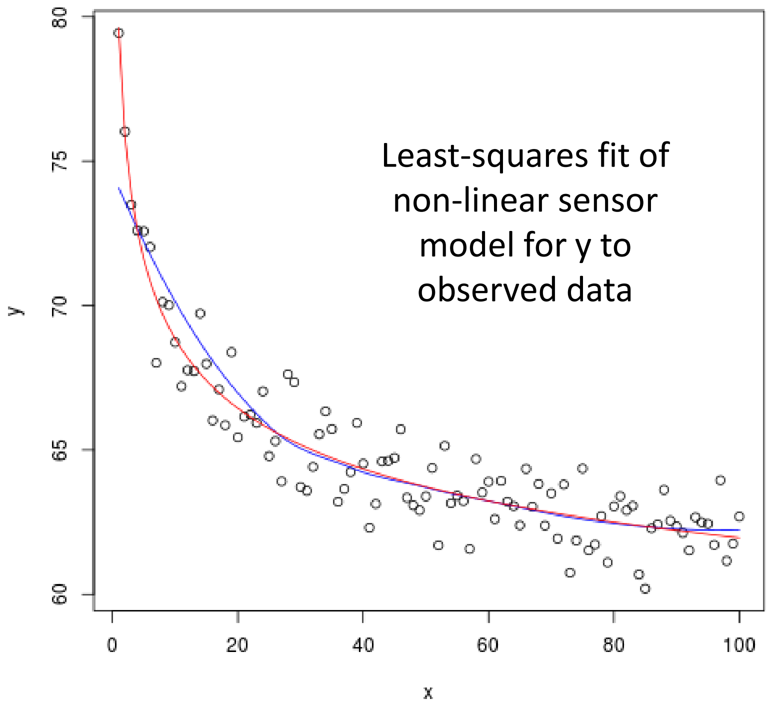
$$\dot{x} = Ax + Bu + v$$
$$y = h(k, B, x) + w$$



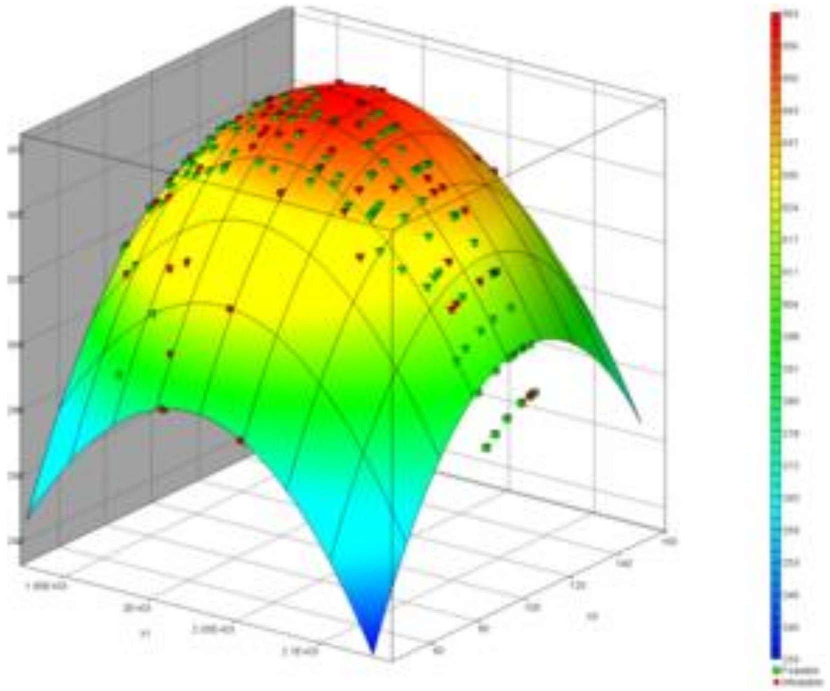
- Also shows up in Kalman filter design
  - Estimate sensor parameters (e.g. IMU accel biases, camera intrinsics/extrinsics)
  - Very slowly changing variables/vector (e.g. tire pressure in car)

Parameter Estimation

- How "good" is our estimate of a parameter or set of parameters?
  - How much data do we need to get "pretty good" estimates?
  - Do we have the "right" model for the data?



$$y = h(k, B, x)$$

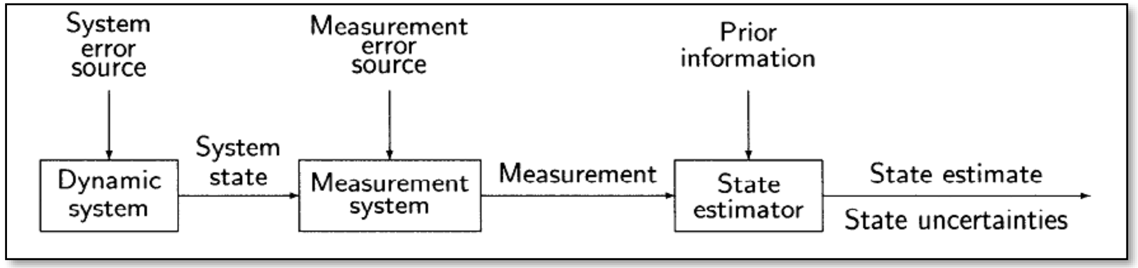


Cost function vs. parameters k and B given y_i and x_i data

Dynamic State Estimation

- What is actual system behavior over time?

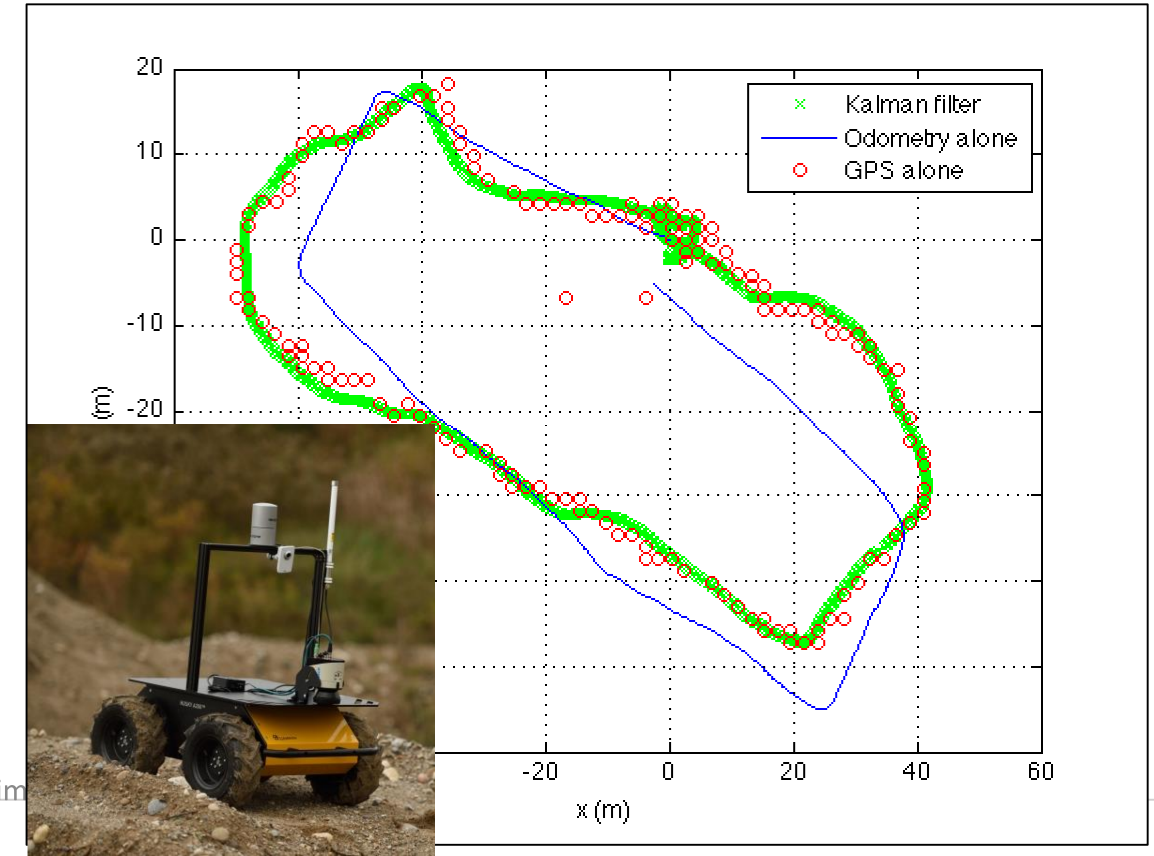
Need models of **dynamic uncertainties (process noise: friction, wind, solar radiation pressure, etc.)** and **noise in sensor data (measurement noise)**



Discrete time: digital computing, relationships to parameter estimation

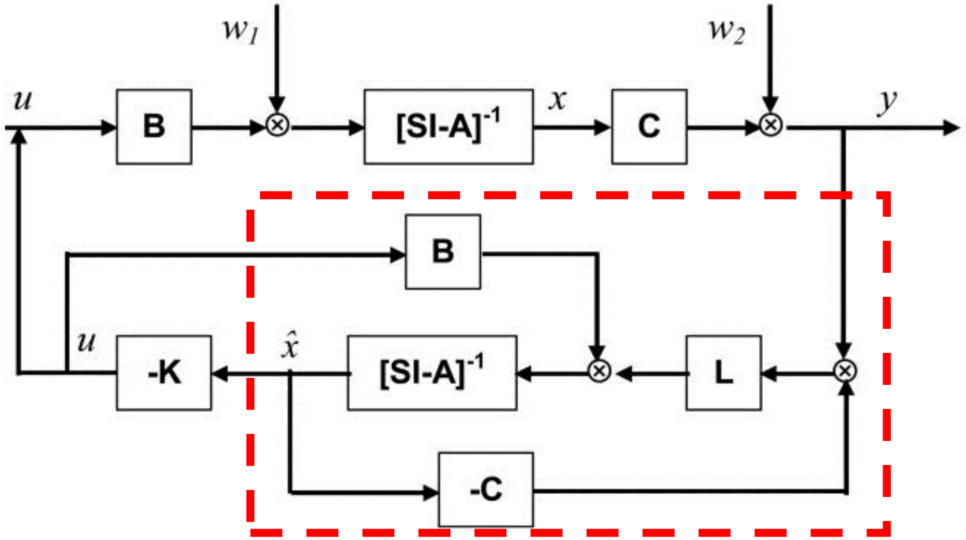
Start with **linear state-space models**:

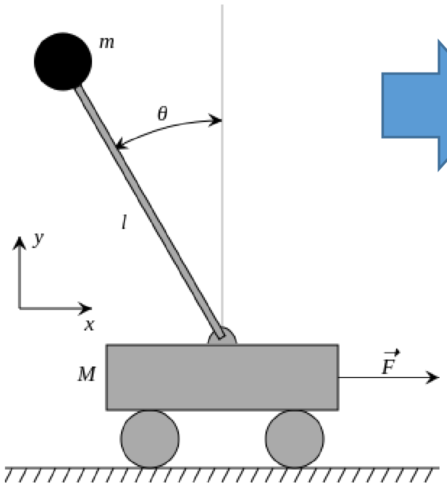
- linear algebra = very powerful tool,
- many unifying insights from control theory (observers), signal processing (stochastic filters), and statistical inference (probability densities)
- set the stage for non-linear methods (EKF; UKF; batch and general ML/Bayes estimators)



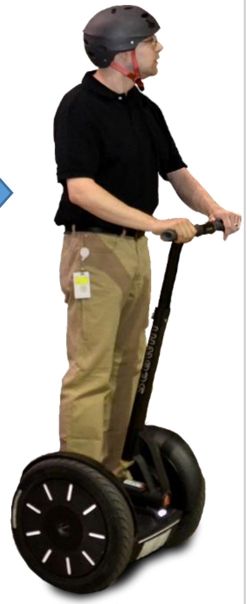
Dynamic State Estimation for Feedback Control

- One of the main drivers for development of Kalman filter in the 1960s





Inverted pendulum



Video: LQR (no Kalman filter) vs. LQG (with Kalman filter) state feedback control of a double inverted pendulum on a cart:
https://www.youtube.com/watch?v=JpNAhKT7yY4

Dynamic State Estimation: The Motion Picture

- Basic 2D Robot EKF SLAM (filtering):

https://www.youtube.com/watch?v=vCVS9WAffi4

- iSAM/SLAM video (U Waterloo)

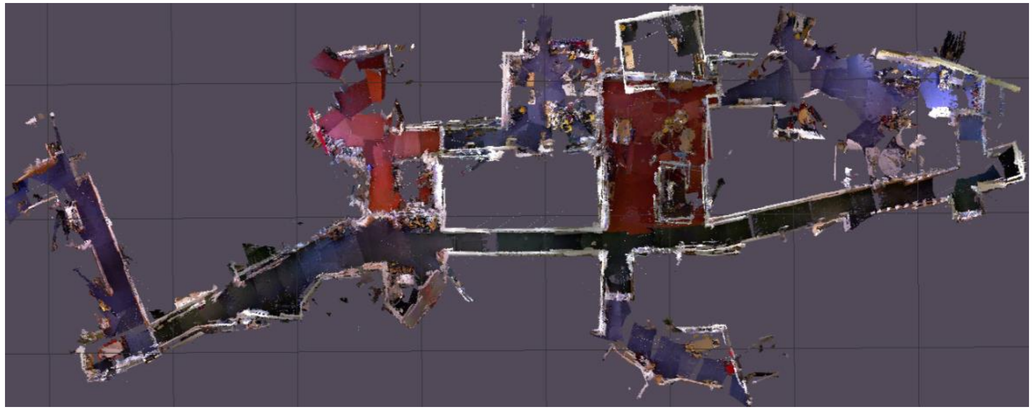
https://www.youtube.com/watch?v=9Y4RQVpp-BY

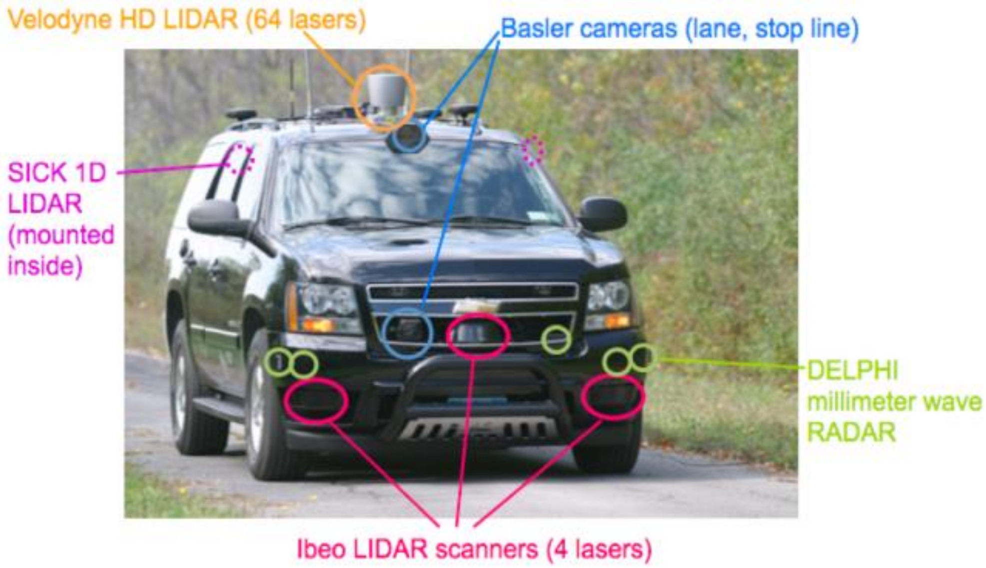
(nonlinear least squares
batch estimation/smoothing)

- Skynet RBPF target tracking videos (folder) (filtering with closed-loop control)

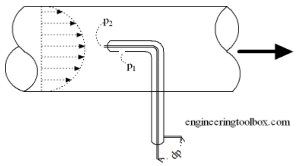
- Stanford power slide parking
(filtering and prediction and control)

http://www.youtube.com/watch?v=gzI54rm9m1Q

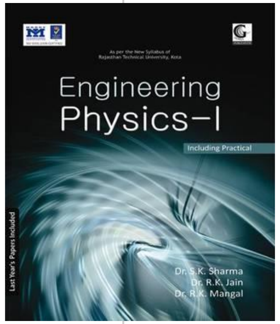


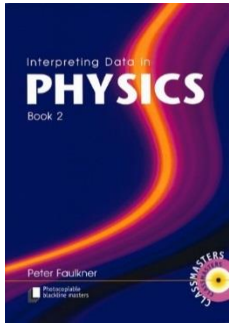


4 Key Parts of Estimation



Observation/
Data



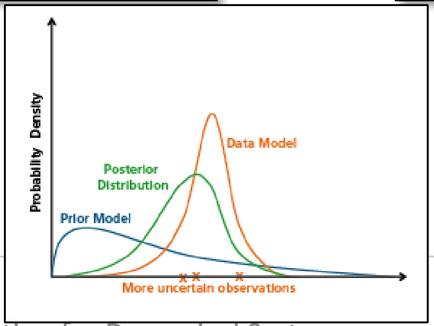


Models → Inference /Fusion → Optimization → $\hat{x} \pm \sigma$

estimate

$$\begin{bmatrix} \dot{x}_1 \\ \vdots \\ \dot{x}_n \end{bmatrix} = \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \cdots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} b_{11} & \cdots & b_{1m} \\ \vdots & \cdots & \vdots \\ b_{n1} & \cdots & b_{nm} \end{bmatrix} \begin{bmatrix} u_1 \\ \vdots \\ u_m \end{bmatrix}$$

$$\begin{bmatrix} y_1 \\ \vdots \\ y_r \end{bmatrix} = \begin{bmatrix} c_{11} & \cdots & c_{1n} \\ \vdots & \cdots & \vdots \\ c_{r1} & \cdots & c_{rn} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} d_{11} & \cdots & d_{1m} \\ \vdots & \cdots & \vdots \\ d_{r1} & \cdots & d_{rm} \end{bmatrix} \begin{bmatrix} u_1 \\ \vdots \\ u_m \end{bmatrix}$$



4 Key Parts of Parameter/State Estimation

- Models tell us what could possibly be true about unknown states, i.e. what uncertainties are out there, how they evolve, and what we should expect to see in our sensor data
    - **Help us constrain our beliefs and guide our reasoning**

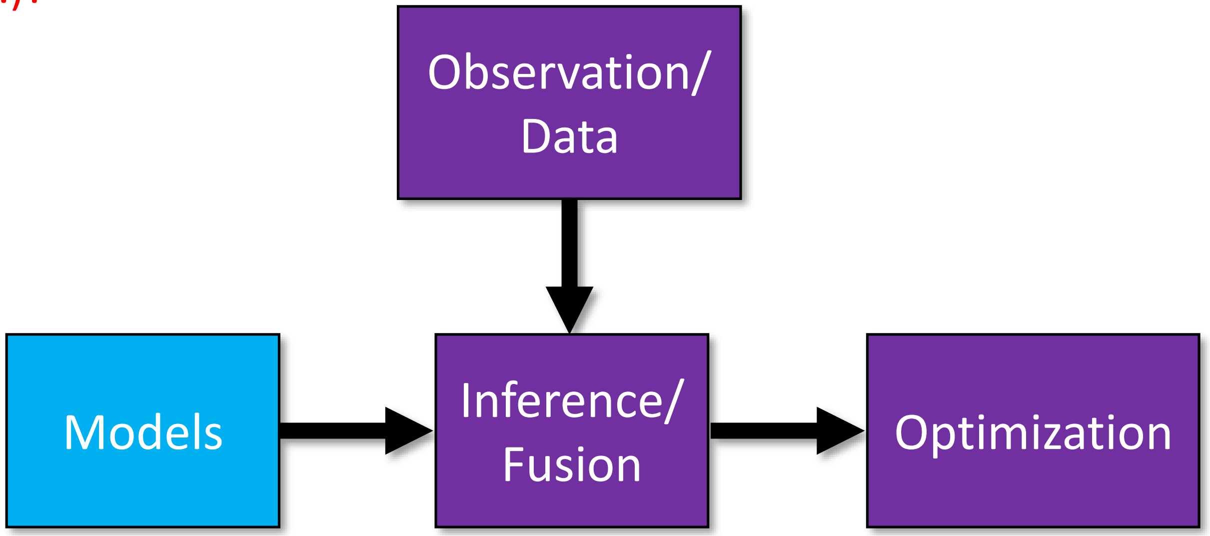
$$\begin{bmatrix} \dot{x}_1 \\ \vdots \\ \dot{x}_n \end{bmatrix} = \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \cdots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} b_{11} & \cdots & b_{1m} \\ \vdots & \cdots & \vdots \\ b_{n1} & \cdots & b_{nm} \end{bmatrix} \begin{bmatrix} u_1 \\ \vdots \\ u_m \end{bmatrix}$$

$$\begin{bmatrix} y_1 \\ \vdots \\ y_r \end{bmatrix} = \begin{bmatrix} c_{11} & \cdots & c_{1n} \\ \vdots & \cdots & \vdots \\ c_{r1} & \cdots & c_{rn} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} d_{11} & \cdots & d_{1m} \\ \vdots & \cdots & \vdots \\ d_{r1} & \cdots & d_{rm} \end{bmatrix} \begin{bmatrix} u_1 \\ \vdots \\ u_m \end{bmatrix}$$

How to model internal dynamical uncertainties ("process noise", e.g. turbulence, friction, leaky valves, stray voltages, grav anamolies, etc.)?

Can we easily model measurement errors (e.g. GPS multipath)? Can we account for all possible sources of error/information?

What to do with very complex systems (e.g. maneuvering targets, chaotic systems, hybrid systems)?



4 Key Parts of Parameter/State Estimation

- Observations tell us what actually happened in the world, given all the possible outcomes for parameters/states
    - need good model of sensors to relate unknowns to what we see/know

How good do sensors have to be?

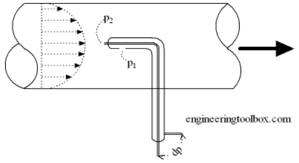
Where to place/point them if limited #?

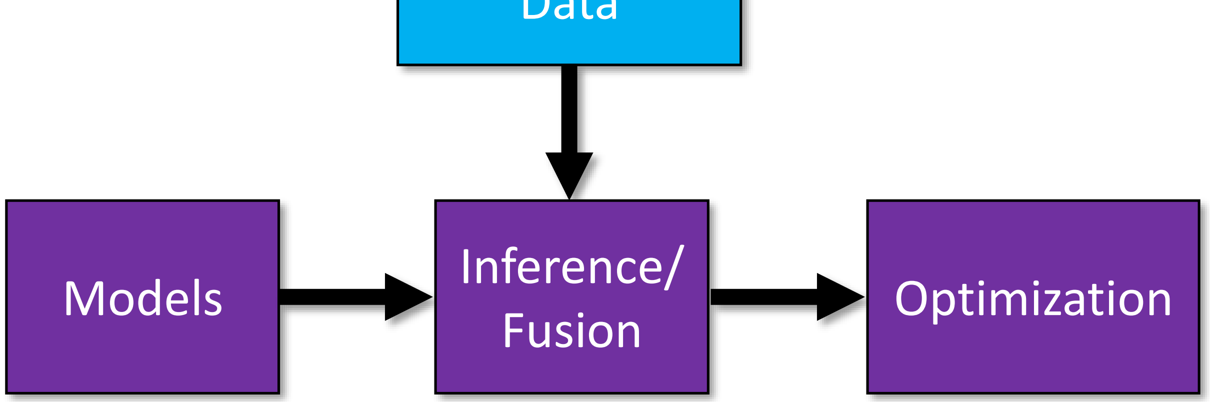
Is information actually independent/new?

How to deal with too much/too little data? Delayed data?

Do we always know where exactly data came from? False alarms/spurious signals?



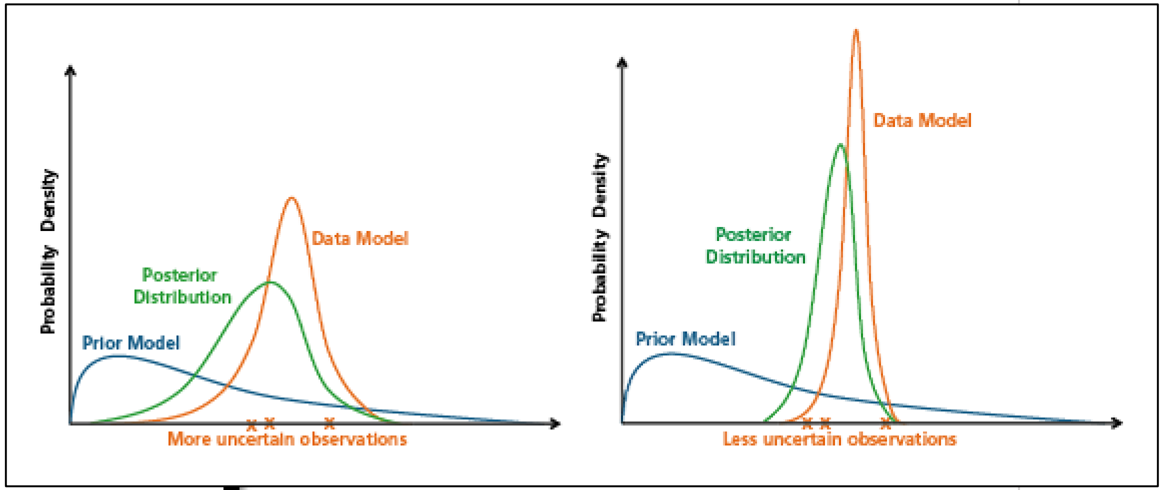



4 Key Parts of Parameter/State Estimation

- Inference tells us **how to update all the possibilities of what could be true about states**, given sensor data we observe and current beliefs
  - **Conditional probabilities:** narrow down possible states based on data

$$P(x|\text{data}) = \frac{P(x)P(\text{data}|x)}{P(\text{data})}$$

"Once you eliminate the impossible, whatever remains, no matter how improbable, must be the truth." – Sherlock Holmes



http://www.air-worldwide.com/Publications/AIR-Currents/2015/A-Martian%E2%80%99s-View-of-Hurricane-Climatology--An-Allegory-for-Bayesian-Learning/

How do we actually find and maintain all these distributions in practice?

How do we deal with "unknown unknowns"?

Models → Inference/ Fusion → Optimization

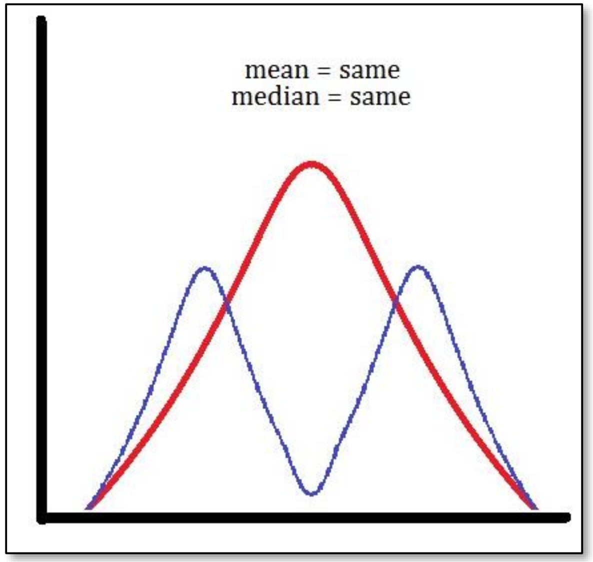
4 Key Parts of Parameter/State Estimation

- Optimization tells us how to come up with a **best guess given our uncertain information** (for all possible states) **and our preferences about guessing wrong**
  - Depends on uncertainties we face and decisions/actions we have to make

How do we define "best"?

What is the estimate being used for?

How sensitive are we to model errors?



Observation/ Data

Models → Inference/ Fusion → Optimization → $\hat{x} \pm \sigma$ estimate

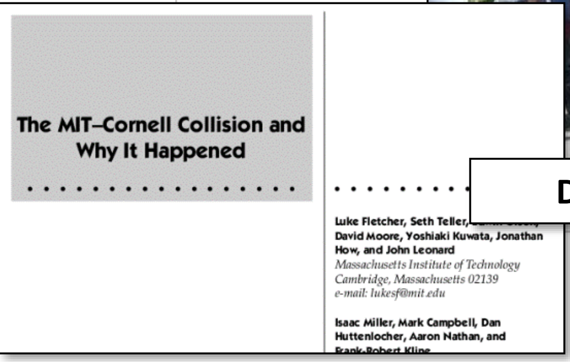
Step 5: Trust Assessment and Validation

- **How much do we believe "our best guess"?** How good can it possibly be? How much information can we pull out of data?
- **How do we know our estimator is behaving sensibly?** Are our models/assumptions correct? Did we catch all relevant uncertainties? Do we have the right data?
- Step 0: does it even make sense to estimate states in first place? (observability…)

Team Cornell's "Skynet"
(NY driver)

Team MIT's "Talos"
(Boston driver)

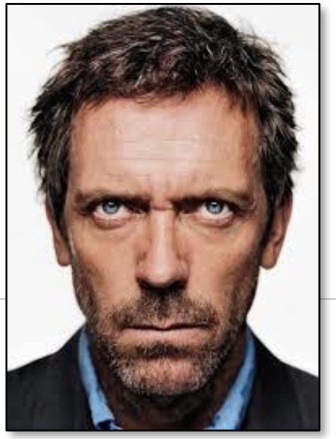
DARPA Urban Challenge 2007



The MIT–Cornell Collision and Why It Happened

Luke Fletcher, Seth Teller,
David Moore, Yoshiaki Kuwata, Jonathan How, and John Leonard
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139
e-mail: lukesf@mit.edu

Isaac Miller, Mark Campbell, Dan Huttenlocher, Aaron Nathan, and Frank-Robert Kline



For Next Time

- Read Introduction and start skimming Chap. 1 of text
  - start dusting off linear algebra and ODEs
  - start dusting off any probability/stats from undergrad

- Extra Credit (+0.001 pts): find the ACTUAL portrait of Thomas Bayes…