Reflections on a Life in Computer Science and Statistics: Norm Matloff

On July 1, 2023, I will retire, after–incredibly–48 years on the faculty of the University of California, Davis.

Just like George Washington :-) , I will give this "Farewell Address." It certainly won't be historical like Washington's, and it's not really a true farewell–I'll have an office, and intend to continue to be active in research and in writing books and software–but I hope some will find it interesting, maybe surprising, and possibly useful.

I'll touch on the following:

- my formative years and early career
  - immigrant connections
  - study at a commuter college
  - grad school
  - on to UCD
  - move from statistics to CS/stat
- research interests and accomplishments
- some "recommendations," on subjects such as:
  - teaching of computer science and statistics
  - the research process and norms
  - research assessment
  - school admissions policies
  - K-12 math education

Formative Years

I was born in Los Angeles, and grew up in East LA and the San Gabriel Valley.

I've lived in immigrant households my entire life, complete with non-English languages. My father was an immigrant from Lithuania, and my parents spoke Yiddish to each other (but spoke English to us kids). My wife is an immigrant from Hong Kong, and Cantonese is the "official language" in our family, including with our daughter.

I attended Cal Poly Pomona as a Math major and an Econ minor. I lived at home, with a 10-minute drive to school, and had a part-time job on campus. Cal Poly was not a "powerhouse" school, but one could get an excellent education if one sought out the right faculty. I had one professor for several Math classes who gave very challenging proofs for homework, and I took the bulk of my Econ classes from one particular professor, whom I still quote to this day.

I took my first programming course as a freshman, 1966. So I've been programming for almost 60 years! I loved it from the start, and find it just as much fun today as I did back then.

After Cal Poly, I joined the PhD program in Math at UCLA, eventually doing a dissertation in abstract probability theory/functional analysis with Tom Liggett. I described both Tom and my days at UCLA here. I also was greatly influenced by Prof. Tom Ferguson and Prof. Olive Jean Dunn; the latter is the one who started me thinking about the inappropriateness of significance testing, about which I've been writing (some would say "harping"!) throughout my career I also talked a lot with Prof. Chuck Stone; he and I later became close friends when we moved to adjoining towns in the Bay Area.

Academic Career

UC Davis!

I had always wanted an academic career, with a very strong preference for California, and was very pleased when the Math Dept. at UCD invited me for an interview as I was finishing my PhD. It was my first-ever plane ride! Though I did start to interview elsewhere, Davis quickly made me an offer, which I accepted. I actually have been a member of four different departments during my years here.

At that time, the Math Dept. had an informal statistics subgroup. We and a couple of others from outside the department formed the Statistics Dept. in 1979. (It was actually called an "intercollegiate division" at that time.)

However, my attention was drawn elsewhere, as the first "computer revolution" was in full swing, and as noted, I had always loved developing software. In addition, frankly, I was getting disillusioned by academic politics. Thus I took a year's leave of absence and worked in Silicon Valley.

Statistics and Computer Science

While on leave, I was under pressure from the Stat Dept. to decide whether to return. I said yes, provided I could have a joint appointment with the Electrical and Computer Engineering (ECE) Dept. (There was no Computer Science Dept. then, and by the way, no separate Engineering salary scale.) The ECE Chair was very supportive, and the campus administration approved the request, and I resumed teaching in Fall 1980.

Later I participated in the establishment of the Division of Computer Science within ECE, which eventually became the Computer Science Dept. in 1989. Again, my strong desire was to have a joint appointment between CS and Stat. Unfortunately, I was told that the two collegiate deans did not want to split a faculty slot, and I would have to choose between one or the other. I chose CS.

My research

I first published two theoretical papers based on my dissertation work, one in the *Annals of Probability* and the other in the *Zeitschrift fur Wahscheinlichskeitstheorie*.

At the same time, I served as a statistical consultant for the Kaiser Permanente medical care system. That experience led directly to my *Biometrika* paper on the asymptotics of averaged regression functions and indeed has influenced much of my subsequent work, to this day.

Since my move to the CS Dept., some of my work has been purely CS, such as in parallel computation. However, much of my work has been in fields in which my stat background has played a major role, such as data security and machine learning.

My work seems to have had some impact. For example, my data security research, such as that published in ACM TODS, led to my being appointed to IFIP WG 11.3, an international UN-sponsored group devoted to fostering work in the field.

Our work on polynomials in neural networks attracted a lot of attention in discussion groups, and has led directly to published work by others, such as this paper in Neural Networks; we started a new field! Similarly, a number of my papers have startedchains of citations (i.e. citations of citations etc.) that continue to the present time. The aforementioned *Biometrika* paper is of that nature.

I've always believed that teaching and research go hand-in-hand. My book, *Statistical Regression and Classification: from Linear Models to Machine Learning*, which was the recipient of the 2017 Ziegal Award, reflects that spirit. Though in the form of the textbook, it contain many "mini-research snippets," material that I believe is new to the field.

Currently I am especially interested in the issue of Fairness in Machine Learning, which combines my computing and stat interests with my lifelong passion for social justice. Two of my recent papers are here and here.

Involvement in controversial issues

Whenever I've seen claims that "1 = 2," and about which I have some new points to add to the discussion, I've spoken out. Sometimes this has been to my detriment, but as a scholar I feel the need to inform and clarify.

In the general sociopolitical realm, I've been active in my support for Affirmative Action in school admissions and hiring. But my most significant impact has been on the H-1B work visa, a program that everyone agrees is badly flawed but for which no good political solution has been agreed upon. My work on that issue led to an invitation by a major law journal to write on the topic. The paper, 99 pages long with over 300 footnotes, has had a major impact, I believe.

My work on these and various other isssues has led to my being quoted or cited in all major news outlets, print and electronic, and testimony to the US House and Senate.

My writings on more academic issues are known to a smaller audience, but are probably even more controversial. As mentioned above, one example is the inappropriateness of significance testing. Another, very narrow but the subject

of especially contentious debate in the field, is my view that the "Tidyverse" approach to teaching the R programming language to noncoders, actually makes things more difficult for that group. Another example is the differential privacy approach to data security; I am a skeptic there too, arguing that the theory is elegant but that the approach is unsatisfactory in many practical situations.

Reflections

As always, I have a few thoughts...

Note first that my comments are meant to be general, not restricted to my own department or university. Of course, my local environment is the one with which I am the most familiar, but I believe I have had enough exposure to other departments/universities to be able to have reasonably informed views.

And of course, there are likely many exceptions to these general remarks.

On Teaching

I've always been excited about teaching, and investigating how students learn. I've been fortunate to win a couple of teaching awards, notably the campus-wide Distinguished Teaching Award. Thugh it must be said that student evaluations of teaching are sometimes out of sync with broader, more nuanced assessment of quality, I am proud that not only I but also several of my department colleagues have won that award over the years.

Class size

We are being asked to teach larger and larger classes. This is rationalized by having "small" discussion sections, but even they are getting larger. The quality of instruction suffers.

One of the many negative consequences of large class sizes is that examinations rely increasingly on rote memorization rather than measurement of true insight. The rote approach is easier to grade, but the message is sent to students that memorization is what counts, not insight.

Begining programming curricula

We are not serving our lower division programming students well. We are stressing syntax and semantics much more than what should be the focus, which is the building of strong programming muscles.

Thus I recommend Python as a first language, due to its not being strongly-typed. Variable types are distractions from developing algorithms.

Programming assignments in an introductory sequence should be progressively more elaborate, with progressively less scaffolding. The latter is key; I tell my students, "If you've never had the experience of sitting down to write code and and thought, 'Whoa, how do I even start?', then you've never had a

chance to develop strong coding skills." Of course, hints should be doled out (sparingly) according to individual student needs, but giving detailed outlines in the assignment statement should be avoided.

We also need to emphasize the importance of software development support infrastructure, such as use of a debugging tool and knowledge of how search paths work. It must be emphasized to students that they must not "erase their brains" of this material after the final exam, but instead should make active use of it in their later courses.

Computer systems issues

People outside the field are amazed when I tell them that most CS students could not answer questions such as "How does a computer boot up?" and "What happens if we run a program with six threads on a four-core machine?" These are not "trade school" questions, but instead rely on fundamental concepts, such as OS being a program, the role of hardware interrupts and so on. I believe that OS and other systems courses should make sure to relate their content to the real world.

Student admissions issues

It's taken as an article of faith among progressives that one must oppose the use of standardized tests for school admissions. If one doesn't toe that line, one is viewed as lacking sympathy for underrepresented minorities. The SAT is correlated with family income, they say, and that is supposed to end the discussion.

I disagree. I regard myself as a progressive, especially in the areas of race, gender and poverty. I am a supporter of Affirmative Action, and in fact chaired our campus Affirmative Action Committee for a couple of years. But the SAT is a valuable tool, and should be retained.

The SAT is far from perfect, but it does contain important information, only some of which is reflected in college grades. Other faculty have reported concrete evidence that student quality has deteriorated subsequent to UCD's dropping the SAT/ACT admissions requirement. They've had to make their courses shallower, ceasing to teach certain topics that worked in the past but now today.

As to the wealth connection, even the critics concede that the effect is very mild, and defenders of the SAT cite other research and the stellar performance of many East Asian or East European children of working-class immigrants, despite poverty.

Aside from equity issues, a major concern is the ever-escalating "arms race" for admissions to the elite schools. Ultrafine scrutiny of grades and test scores makes no sense, and frankly I see a lot of gaming the system.

Accordingly, I have long advocated a random lottery for school admissions, in which a threshold would be set based on test scores and grades, and admission being done randomly from the resulting pool. The threshold would be set at

a level deemed necessary for doing good work at that institution. This would ameliorate the "arms race" problem, I believe, and would recognize that numerics are imperfect measures at best.

"Kids today"

Ever since Socrates, teachers have complained about reduced student quality. One must also treat such claims with a healthy dose of skepticism.

But I think there is something much more fundamental–a declining willingness to think about some topic *really intensely*, for an extended period of time. University students have always treated Teaching Assistants asa First Resort rather than a Last one, essentially expecting full solutions from a TA rather than the TA merely pointing them in the right direction. Many instructors implicitly encourage such attitudes. I regard all of this as a serious problem for our economy, society and so on.

K-12 math education

In the last few years there has been an alarming trend in US K-12 math education, especially in big cities, of diluting curricula in the name of "equity." The result of course has been that the very children this supposedly is aimed to help, our most vulnerable, are harmed instead. As a lifelong passionate advocate of improving conditions for the underserved in our society, this greatly pains me.

Two aspects are especially concerning:

- The push to have students take "data science" instead of Algebra 2: I put the term in parentheses, because the courses, such as the one used in the LA Unified School District, are very thin. Even worse, the notion that "data science" could be taught to students who are not familiar with the slope of a line is sadly absurd.
- The de-emphasis on number facts, the 6+9 and 5 X 8 nuts and bolts: I've always believed that education should emphasize insight rather than rote memorization, but facility with number facts is crucial for having "a feel" for numbers–the large and the small, approximate computation and so on. We're producing students who won't know they're being shortchanged by a store cleark!

This sometimes has resulted in the bizarre. Consider this interview of Sal Khan, founder of the Khan Academy. Khan finds the kids can do an intricate algebraic operation yet then need a calculator to calculate 3 X 7 to complete it!

The San Francisco Unified School District (SFUSD) has been implementing such "reforms" the last few years, with disastrous results in the statewide math tests: Scores for low-socioeconomic status (low-SES) children declined after the new curriculum was implemented, while scores for low-SES California kids statewide were rising.

On Research

At least in Computer Science, we are in a crisis:

- The quality of reviewing, especially in conferences, is often atrocious. Reviewers have tunnel vision as to what kind of work is important, and often a reviewer simply does not have the expertise/insight to do a proper review. For example, in the case of paper I reviewed in 2023 for an ML conference, I felt the work was good but did not go far enough; I gave it a Weak Accept rating. The second reviewer merely said, "Not enough examples," and rated the work as Strong Reject, which in my view stemmed from the reviewer's not having the background needed to assess the work. Frankly, the reviewer's statement about examples was just an excuse, I believe. In my experience, that kind of incident occurs quite frequenty. Journal reviewing is somewhat better. I would recommend de-emphasizing conference work.

- There's often too much emphasis on quantity instead of quality. The old joke that "Deans can't read but they can count" seems to apply more and more to faculty voting on advancements as well. There have been proposals that in review of faculty in advancement cases, the candidate submits his/her 5 (or whatever number) best papers for consideration. I believe such a policy is urgently needed.

- It appears to me, based both in the case of my own department and some observations of others, there is less intradepartmental (as opposed to *inter*departmental) research collaboration in CS than, say, in Statistics.

  In my own case, I had hoped to be the "resident statistician" in CS, leading to a lot of collaboration with my CS colleagues. This turned out not to be the case–I have not had a paper coauthored with a department colleague since 2009–and I see rather little joint work among other colleagues.

  This may be related to the CS emphasis on funded research. Whatever the reason, it results in many missed opportunities.

On the Dark Side of Academia

When I first began at Davis, a senior colleague advised me, "Form over substance, that's how the system works." And nearly a century ago, when President Herbert Hoover was asked why he left Stanford to serve as President,' he joked, "Because I wanted to get out of politics."

If anything, it's worse today. Actions are often taken that ostensibly improve teaching or research, while actually having no real impact, or even a negative one. And too many committees decide on the basis of personal or other extraneous conditions rather than what is good for the university students and faculty. Quite often there is an appalling lack of transparency.

I served on the academic Senate Privilege and Tenure Committee, including two years as chair. I saw many cases of egregiously unfair treatment, sometimes vindictive, often heartbreaking. And there are many, many cases that, though less extreme in severity and not presented to the committee, have major adverse impacts on the victims. In some cases, such machinations by faculty are done in collusion with department office staff.

It's unclear how to solve this problem in general, but improved transparency would be at the top of the list of remedies.

On the future impact of Large Language Models

My view on this is mostly positive.

I believe they are currently less useful for teach matter than is generally thought. ChatGPT is pretty good at writing code, but often lacking in being able to "see the big picture." See my essay, (Like College Students, ChatGPT Finds Statistics Difficult)[https://magazine.amstat.org/blog/2023/04/01/chatgpt-finds-statistics-difficult/]. Version 4 seems equally weak.

Thanks

I've been blessed with a wonderful family, whose love, humor and wisdom are inspiring and humbling. I continue to believe that being an academic is the best job in the world. I've been extremely fortunate to be in this line of work, and I thank everyone who has made it possible. My heartfelt thanks to all!