LECTURE 5: MULTI-ARMED BANDITS AND OTHER COMPLEX EXPERIMENTS

Guido Imbens – Stanford University

Economics 272, GSB 507, Spring 2025

OUTLINE

1. Introduction
2. Multi-armed Bandits
   - Thompson Sampling
   - Upper Confidence Bounds
   - Exploration Sampling
3. Multiple Randomization Designs

INTRODUCTION

- Steve Levitt, Freakonomics:

"Or, put another way, anyone can do program evaluations based on true randomization, so why should some of the world's best economists be devoting so much of their time to such exercises? The great economists should be trying to do something that is harder."

INTRODUCTION

- Steve Levitt, Freakonomics:

  "Or, put another way, anyone can do program evaluations based on true randomization, so why should some of the world's best economists be devoting so much of their time to such exercises? The great economists should be trying to do something that is harder."

- Not quite so easy!

  Lots of interesting complex experimental designs these days.

INTRODUCTION

- Suppose we want to evaluate a new treatment, *e.g.*, a new search algorithm.
- We can do a randomized experiment, $N = 1000$ units (*e.g.,* search queries), N_T treated, $N_C = N - N_T$ controls, estimator

$$\hat{\tau} = \overline{Y}_T - \overline{Y}_C, \qquad s.e. = \sqrt{S_T^2/N_T + S_C^2/N_C}$$

- We could then decide whether to implement the new treatment or not, based on significance level,

$$\hat{\tau} > 1.96 \times s.e.$$

- Or (better!) use prior distribution for τ and make Bayesian decision.

CHALLENGE

- Now suppose we want to evaluate $K = 10$ new search algorithms. We assign N/K units to each of them, and get estimates and standard errors for each treatment:

$$\hat{\mu}_k = \overline{Y}_k, \qquad s.e.(\hat{\mu}_k) = \sqrt{S_k^2/N_k}.$$

- How do we use these numbers to make a decision?
- We could test various null hypotheses

$$H_0 : \mu_k = \mu_m, \qquad vs \ H_a : \mu_k \neq \mu_m,$$

  taking account of multiple testing
- That does not answer the right question.

WHAT IS THE RIGHT QUESTION?

- What we want to do is make a good decision.
- Decision: after the experiment, go with treatment k if

$$\overline{Y}_k = \max_{m=1}^{K} \overline{Y}_m.$$

- Why would that not be a good decision rule? Is it
  - Risk aversion?
  - Heterogenous cost of treatment?
  - Prior beliefs about efficacy?
  - Maybe ok if standard errors for each $\overline{Y}_k$ are similar, but certainly not otherwise.

EXPERIMENTAL DESIGN PROBLEM

- Consider a second problem. Initially we assign M/K units to each treatment, run the experiment, observe $\overline{Y}_k$ and $\hat{\sigma}_k = s.e.(\overline{Y}_k)$.
- We want to do a second experiment, where we assign the next M units to the K treatments.
- After that second experiment we stop and choose the optimal treatment, which will be based on the maximum value of the $\overline{Y}_k$.
- How should we allocate the second batch of M units to the K treatments?
- We should clearly not:
  - Assign equally to all K treatments (wasting units on inferior treatments).
  - Assign all the currently best performing treatment (not learning anything anymore).

SIMPLE EXAMPLE

- Suppose there are K treatments, outcome is binary, $Y_i(k) \sim \mathcal{B}(1, \mu_k)$.
- We are interested in identifying the treatment arm k^* with the highest value of μ_k.
- Suppose we start by observing 100 draws for each arm, and get $\hat{\mu}_k$ for each arm. Then our best guess for k^* is the arm with the highest $\hat{\mu}_k$.
- Now suppose we have the opportunity to allocate another 100 units to these K treatment arms, how should we do that?
- Suppose initially, $\hat{\mu}_1 = 0.10$, $\hat{\mu}_2 = 0.80$, $\hat{\mu}_3 = 0.81$, $\hat{\mu}_4 = 0.70$

SIMPLE EXAMPLE (CTD)

- General observations:
  - Allocating a lot of units to treatment arm 1 does not serve much of a purpose: it is unlikely that arm 1 is the best arm.
  - To learn about the optimal arm, we should assign more units to treatment arms 2, 3 and 4.
  - But: how many units to each arm?
  - Should we assign a lot of, or any, units to arm 4?
- We need to balance exploration from assigning to currently not optimal arms and exploitation from assigning to currently best guess for optimal arm.

TWO METHODS: THOMPSON SAMPLING AND UPPER CONFIDENCE BOUND METHODS

- Two approaches to determining assignment for next unit.
- In both cases we assign more units to arms that look promising, in slightly different ways:
  - Thompson sampling: Use Bayesian approach where we calculate posterior probability that arm k is the optimal arm, and assign to this arm with probability proportional to the probability that it is the best arm.
  - Upper Confidence Bound method: calculate confidence intervals for each μ_k, with confidence level α_T (T is the total sample size for entire experiment), $\alpha_T \to 1$ as $T \to \infty$.

THOMPSON SAMPLING: INFORMAL

- Four Steps:
  1. Collect an initial batch of data, with N units assigned with equal probability to each of the K arms.
  2. Given the success frequencies $\hat{\mu}_k$ for each arm, and given prior distribution for μ_k (say flat prior), calculate the posterior distribution of $\mu_1, \ldots, \mu_K$. (Easy here because these are Beta posteriors)
  3. Allocate the next unit to arm k with probability equal to the probability that $\mu_k = \max_{m=1}^{K} \mu_m$. (Easy to implement by simulation.)
  4. Update the posterior distribution given the new observation and use that for assigning the next unit.

THOMPSON SAMPLING: INTERPRETATION

- As you go along, you will increasingly assign units to the arms that have high success probability of being the optimal arm.
- This balances exploration: learn more about all arms by allocating units to them, and exploitation: send more units to arms that do well.
  - exploration is about learning about things that you are still uncertain about, and that may not be optimal.
  - exploitation is about focusing on treatment arms that given current information look good.
- In example, arm 1 does very poorly, don't send more units to that arm. We are not sure about the other arms, so we send units to all of them, but more to 2 and 3 than to 4.
- Very effective way to experiment in settings with many treatments, and with sequential assignment.

THOMPSON SAMPLING, FORMAL ANALYSIS: SET UP

- K arms, T periods, one unit per period (typically batches of units, multiple units per period, but no conceptual difficulty with that)
- $A_t \in \{1, \ldots, K\}$ is the action (arm choosen) in period t.
- Given action $A_t = k$, the reward in period t is $Y_t \sim \mathcal{B}(1, \mu_k)$ (binary for simplicity).
- Optimal arm is $k^* = \arg\max_{k=1}^{K} \mu_k$, with expected reward $\mu^* \equiv \mu_{k^*}$
- A policy is a function $\pi(A_1, Y_1, \ldots, A_{t-1}, Y_{t-1})$ that maps past rewards and actions into the action space.
- The goal is to maximize the average (or cumulative) reward $\sum_{t=1}^{T} Y_t/T$.

THOMPSON SAMPLING, ALGORITHM: REGRET

- How do we compare different algorithms/policies?
- If we follow the optimal policy k^*, our reward is random with mean μ^*.
- Given our policy $\pi(\cdot)$ our reward is also random with a more complex function.
- The regret of a particular policy $\pi(\cdot)$ is the expected difference in rewards, over the experiment, of following the optimal policy (k^*) versus that particular policy $\pi(\cdot)$

$$R(\pi(\cdot)) = \mu^* - \mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T} Y_t \,\middle|\, \pi(\cdot)\right]$$

- We consider the regret for large T, and would like to choose a policy to (approximately) minimize regret.

THOMPSON SAMPLING, ALGORITHM: SET UP

- Model $Y_t|A_t = k \sim \mathcal{B}(1, \mu_k)$
- Prior: $\mu_k \sim Beta(\alpha_k(1), \beta_k(1))$

  (in practice typically $\alpha_k(1) = \beta_k(1) = 1$ to get flat (uniform) prior on $[0, 1]$)
- Consider period t:
  - Given arm k is chosen in period t,
  - if the prior for μ_k at the beginning of period t is $Beta(\alpha_k(t), \beta_k(t))$,
  - and if the reward is Y_t,

  then the posterior for μ_k after period t is

$$Beta\Big(\alpha_k(t) + Y_t, \beta_k(t) + (1 - Y_t)\Big)$$

PSEUDO CODE FOR THOMPSON SAMPLING FOR BERNOULLI BANDITS WITH FLAT PRIORS

for $k = 1$ to K **do**

  $\alpha_k(1) = 0$, $\beta_k(t) = 0$

end for

for $t = 1$ to T **do**

  for $k = 1$ to K **do**

    Sample $\mu_k(t)$ from $\mathrm{B}(\alpha_k(t), \beta_k(t))$

  end for

  Play arm $k(t) = \arg\max_k \mu_k(t)$ and observe reward Y_t

  If $Y_t = 1$, then $\alpha_{k(t)}(t+1) = \alpha_{k(t)}(t) + 1$, $\beta_{k(t)}(t+1) = \beta_{k(t)}(t)$, if $Y_t = 0$, then $\alpha_{k(t)}(t+1) = \alpha_{k(t)}(t)$ $\beta_{k(t)}(t+1) = \beta_{k(t)}(t) + 1$.

  for $K = 1$ to k **do**

    if $k \neq k(t)$, update $\alpha_k(t+1) = \alpha_k(t)$, $\beta_k(t+1) = \beta_k(t)$.

  end for

end for

FORMAL ANALAYSIS: REGRET BOUNDS

- How do we think about the properties of this and other algorithms?
- Minimize Regret. Regret is nonnegative, only zero if we choose k^* at every stage, but that is not achievable.
- With time to learn (T large), we should get close to zero regret, .
- Lai and Robbins (1985) give a lower bound on the regret, for large T:

$$\mathbb{E}\left[R(\pi(\cdot))\right] \geq \frac{\ln(T)}{T} \sum_{k \neq k^*} \frac{\mu^* - \mu_k}{I(\mu_k, \mu^*)} + o(1)$$

$$I(\mu_k, \mu_m) = (1-\mu_k) \ln\left(\frac{1-\mu_k}{1-\mu_m}\right) + \mu_k \ln\left(\frac{\mu_k}{\mu_m}\right) \qquad \text{Kullback – Leibler divergence}$$

- Interpretation:
  - Regret goes down proportional to $\ln(T)/T$.
  - Regret is small if the μ_k differ substantially from the optimal μ^*
  - Hard to learn if some μ_k are close to μ^*.

FORMAL ANALYSIS: REGRET BOUND FOR THOMPSON SAMPLING

- Expected Regret of Thompson Sampling algorithm:

$$\mathbb{E}\left[R(\pi(\cdot))\right] = O\left(\frac{\ln(T)}{T}\sum_{k\neq k^*}\frac{1}{\left(\mu_K^* - \mu_k\right)^2}\right)$$

  (Agrawal & Goyal, 2012, Thm 2)

- Algorithm is not optimal: in period T you should always pick the arm with the highest posterior mean for μ_k, and not randomize. (There is no benefit to learning anymore at that stage, because you cannot put the additional learning to any use.)
- But close to optimal, and easy to implement.

UPPER CONFIDENCE BOUNDS METHODS

- Assign a first batch of units to K treatment arms randomly.
- Construct confidence intervals with confidence level α_T for the success rate μ_k for each treatment arm. *E.g.,* start with $\alpha_T = 1 - 1/T^2$.
- So for the first arm the confidence interval for μ_1 may be $[0.70, 0.80]$, for μ_2 $[0.65, 0.75]$ and for μ_3 $[0.72, 0.82]$.
- Assign the next unit to the treatment with the highest value for the upper confidence bound, arm 3 in this case.
- Then given the outcome for the next unit, update the confidence intervals for the chosen arm. Now the new confidence intervals may be $[0.70, 0.80]$, $[0.65, 0.75]$ and $[0.68, 0.77]$, and so now the next unit is assigned to treatment 1.

FORMAL ANALAYSIS UPPER CONFIDENCE BOUNDS METHODS: SET UP

- $T(k,t)$ is number of times arm k has been played in first t periods.
- $\hat{\mu}(k,t)$ is average reward for arm k after t periods (average of rewards for the $T(k,t)$ times the arm was played.
- $UCB(k,t)$ is upper confidence bound for arm k after period t, equal to $\hat{\mu}(k,t) + g(\alpha)/\sqrt{T(k,t)}$, where $1 - \alpha$ is confidence level, eg $\alpha = 0.95$. Recall that the confidence bounds scale by the square root of the sample size. Typically $g(p) = \sqrt{2\ln(1/p)}$.
- p is function of total experiment size, *e.g.* $\alpha = 1 - 1/T^2$. (If the experiment is long, you want to do more exploration, and arms that have a low $\hat{\mu}(k,t)$ can still be chosen because you look for very optimistic scenarios.)

PSEUDO CODE FOR UPPER CONFIDENCE BOUND ALGORITHM

1: **for** $k = 1$ to K **do**

2: $UCB(k,1) = \infty$, $T(k,1) = 0$, $\hat{\mu}(k,1) = 0$.

3: **end for**

4: **for** $t = 1$ to T **do**

5: Play arm $k(t) = \arg\max_k UCB(k,t)$ and observe reward Y_t.

6: update information for arm $k(t)$

$$\hat{\mu}(k(t), t+1) = \frac{T(k(t),t) \times \hat{\mu}(k(t),t) + Y_t}{T(k(t),t)+1} \qquad T(k(t), t+1) = T(k(t),t) + 1$$

$$UCB(k(t), t+1) = \hat{\mu}(k(t), t+1) + \sqrt{\frac{2\ln(1/(1-\alpha))}{T(k(t), t+1)}}$$

7: **for** $K = 1$ to k **do**

8: For k$\neq k(t)$, update $\hat{\mu}(k, t+1) = \hat{\mu}(k,t)$, $T(k, t+1) = T(k,t)$, $UCB(k, t+1) = UCB(k,t)$.

FORMAL ANALAYSIS: REGRET BOUND UPPER CONFIDENCE BOUND ALGORITHM

- Expected Regret of UCB algorithm:

$$\mathbb{E}\left[R(\pi(\cdot))\right] \leq \frac{8\ln(T)}{T}\sum_{k\neq k^*}\frac{1}{\mu_K^* - \mu_k} + \frac{1+\pi^2/3}{T}\sum_{k\neq k^*}(\mu^* - \mu_k)$$

(Auer, Cesa-Bianchi & Fisher, 2002)

CHALLENGES AND EXTENSIONS: INFERENCE

- Consider a very simple bandit-type experiment.
- We have two batches of 100 units each. In the first batch we assign 50 units to each of two treatments, 0 and 1.
- Let

$$\hat{\mu}_{t,w} = \frac{\sum_i 1\{W_{it} = w\} Y_{it}}{\sum_i 1\{W_{it} = w\}} \qquad \text{and} \qquad \hat{\mu}_w = \frac{\sum_{i,t} 1\{W_{it} = w\} Y_{it}}{\sum_{i,t} 1\{W_{it} = w\}}$$

  be the average outcome for batch t and treatment w, and the average by treatment status, respectively.
- In the second batch we assign 75 units to the treatment with the highest value of $\mu_{1,w}$ and 25 units to the other treatment.

CHALLENGES AND EXTENSIONS: INFERENCE FOR THOMPSON SAMPLING AND UCB METHODS

- Consider the expectation of $\hat{\mu}_{t,w}$:

$$E[\hat{\mu}_{t,w}] = \mu_w$$

- Now consider the expectation of $\hat{\mu}_w$:

$$E[\hat{\mu}_w] = E\left[\frac{\sum_{i,t} 1\{W_{it} = w\}Y_{it}}{\sum_{i,t} 1\{W_{it} = w\}}\right]$$

- Is that unbiased for μ_w?
- Inference is a challenge!

CHALLENGES AND EXTENSIONS: CONTEXTUAL BANDITS

- Suppose we also have covariates X_i for each unit, and want to find the function that assigns each unit to the treatment with the highest expected return as a function of the covariates.
- Given a parametric model for the expected return, we can directly use Thompson sampling.
- But: that may give you false confidence. Suppose you get a first batch with values for X_i between 0 and 1. The parametric model may tell you that for $X_i = 2$ you should always send a unit to treatment 0 rather than 1, even if you have never seen units with similar values for X_i, simply based on extrapolation.
- So, you may want to explore more than your parametric model suggests.

CHALLENGES: BEST ARM IDENTIFICATION

- Standard bandit algorithms balance exploration and exploitation
- In a simple setting with two treatments these algorithms will eventually settle most of the time on a single treatment (the one with the best outcome).
- Suppose we want to just focus on exploration and learn the best arm, without caring about the regret for the units in the experiment.
- In the two arm case, we should keep sending equal numbers of units to the two treatments, no matter how much we have learned already about which arm is better.

CHALLENGES AND EXTENSIONS: BEST ARM IDENTIFICATION

- Recall regret

$$R(\pi(\cdot)) = \mu^* - \mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T} Y_t \middle| \pi(\cdot)\right]$$

- We can change the objective function. After the T–period experiment we choose the policy that maximizes the posterior mean:

$$\hat{k} = \arg\max \mathbb{E}[\mu_k | A_1, Y_1, \ldots, A_T, Y_T]$$

- The regret going forward will be

$$R'(\pi(\cdot)) = \mu^* - \mu_{\hat{k}}$$

- How does that change the policies?
- Exploration Sampling (Kasy and Sautmann, 2021) is a simple modification of Thompson Sampling. (other modifications are available, also for UCB)

PSEUDO CODE FOR EXPLORATION SAMPLING FOR BERNOULLI BANDITS WITH FLAT PRIORS

for $k = 1$ to K **do**

  $\alpha_k(1) = 0$, $\beta_k(t) = 0$

end for

for $t = 1$ to T **do**

  for $k = 1$ to K **do**

    Sample $\mu_k(t)$ from $\mathrm{B}(\alpha_k(t), \beta_k(t))$

  end for

  Play arm $k(t) = \arg\max_k\{(1 - \mu_k(t)) \times \mu_k(t)\}$ and observe reward Y_t

  If $Y_t = 1$, then $\alpha_{k(t)}(t+1) = \alpha_{k(t)}(t) + 1$, $\beta_{k(t)}(t+1) = \beta_{k(t)}(t)$, if $Y_t = 0$, then $\alpha_{k(t)}(t+1) = \alpha_{k(t)}(t)$ $\beta_{k(t)}(t+1) = \beta_{k(t)}(t) + 1$.

  for $K = 1$ to k **do**

    if $k \neq k(t)$, update $\alpha_k(t+1) = \alpha_k(t)$, $\beta_k(t+1) = \beta_k(t)$.

  end for

end for

MULTIPLE RANDOMIZATION DESIGNS

- Typically the assignment is a vector
- Sometimes we can think of treatment assignment as matrix

$$\mathbf{W} = \begin{pmatrix} \textit{viewers} \rightarrow & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ \textit{movies} & & & & & & & & \\ \downarrow & & & & & & & & \\ 1 & ? & ? & ? & ? & ? & ? & ? & ? \\ 2 & ? & ? & ? & ? & ? & ? & ? & ? \\ 3 & ? & ? & ? & ? & ? & ? & ? & ? \\ 4 & ? & ? & ? & ? & ? & ? & ? & ? \\ 5 & ? & ? & ? & ? & ? & ? & ? & ? \end{pmatrix}$$

MULTIPLE RANDOMIZATION DESIGNS

Movie Exper.
Random. Movies

$$\mathbf{W} = \begin{pmatrix} \downarrow \textit{movies} \quad \textit{viewers} \rightarrow & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 1 & C & C & C & C & C & C & C \\ 2 & C & C & C & C & C & C & C \\ 3 & T & T & T & T & T & T & T \\ 4 & T & T & T & T & T & T & T \end{pmatrix}$$

Viewer Exper.
Random. Viewers

$$\mathbf{W} = \begin{pmatrix} \downarrow \textit{movies} \quad \textit{viewers} \rightarrow & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 1 & C & T & C & C & T & T & T \\ 2 & C & T & C & C & T & T & T \\ 3 & C & T & C & C & T & T & T \\ 4 & C & T & C & C & T & T & T \end{pmatrix}$$

MULTIPLE RANDOMIZATION DESIGNS

- But we can do more interesting things than movie or viewer experiments:
- Simple Multiple Randomization Design

$$\mathbf{W} = \begin{pmatrix} \textit{viewers} \rightarrow & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ \textit{movies} & & & & & & & & \\ \downarrow & & & & & & & & \\ 1 & C & C & C & C & C & C & C & C \\ 2 & C & C & C & C & C & C & C & C \\ 3 & C & C & C & C & T & T & T & T \\ 4 & C & C & C & C & T & T & T & T \\ 5 & C & C & C & C & T & T & T & T \end{pmatrix}$$

- Three comparison groups that are *ex ante* comparable, but *ex post* possibly different: C, C, C
- These comparisons tell us about spillovers.

MULTIPLE RANDOMIZATION DESIGNS

- We can compare average outcomes in each of the four groups:

$$\overline{Y}_{T},\ \overline{Y}_{C},\ \overline{Y}_{C},\ \overline{Y}_{C}$$

  - $\overline{Y}_{T} - \overline{Y}_{C}$ tells us about total effect.
  - $\overline{Y}_{C} - \overline{Y}_{C}$ tells us about indirect effect of within-viewer/across-movie spillovers.
  - $\overline{Y}_{C} - \overline{Y}_{C}$ tells us about indirect effect of within-movie/across-viewer spillovers.
  - $\overline{Y}_{T} - \overline{Y}_{C} - \overline{Y}_{C} + \overline{Y}_{C}$ tells us about direct effect.

MORE COMPLEX MULTIPLE RANDOMIZATION DESIGNS

More complex Multiple Randomization Design:

$$\mathbf{W} = \begin{pmatrix} & \multicolumn{8}{c}{\textit{Movie Ex periment}} & \multicolumn{5}{c}{\textit{Viewer Ex periment}} \\ \textit{viewers} \rightarrow & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 \\ \textit{movies} & A & A & A & A & A & A & A & A & B & B & B & B & B \\ \downarrow & & & & & & & & & & & & & \\ 1 & C & C & C & C & C & C & C & C & T & C & T & C & C \\ 2 & C & C & C & C & C & C & C & C & T & C & T & C & C \\ 3 & T & T & T & T & T & T & T & T & T & C & T & C & C \\ 4 & C & C & C & C & C & C & C & C & T & C & T & C & C \\ 5 & T & T & T & T & T & T & T & T & T & C & T & C & C \\ 6 & T & T & T & T & T & T & T & T & T & C & T & C & C \\ 7 & C & C & C & C & C & C & C & C & T & C & T & C & C \end{pmatrix}$$

REFERENCES

- Agarwal, Alekh, Daniel Hsu, Satyen Kale, John Langford, Lihong Li, and Robert Schapire. "Taming the monster: A fast and simple algorithm for contextual bandits." In International Conference on Machine Learning, pp. 1638-1646. PMLR, 2014.
- Agrawal, Shipra, and Navin Goyal. "Analysis of thompson sampling for the multi-armed bandit problem." In Conference on learning theory, pp. 39-1. JMLR Workshop and Conference Proceedings, 2012.
- Auer, Peter, Nicolo Cesa-Bianchi, and Paul Fischer. "Finite-time analysis of the multiarmed bandit problem." *Machine learning* 47 (2002): 235-256.
- Bajari, Patrick, Brian Burdick, Guido W. Imbens, Lorenzo Masoero, James McQueen, Thomas S. Richardson, and Ido M. Rosen. "Experimental design in marketplaces." *Statistical Science* 38, no. 3 (2023): 458-476.
- Dimakopoulou, Maria, Zhengyuan Zhou, Susan Athey, and Guido Imbens. "Estimation considerations in contextual bandits." arXiv preprint arXiv:1711.07077 (2017).

REFERENCES (CTD)

- Dimakopoulou, Maria, Zhengyuan Zhou, Susan Athey, and Guido Imbens. "Estimation considerations in contextual bandits." arXiv preprint arXiv:1711.07077 (2017).
- Kasy, Maximilian, and Anja Sautmann. "Adaptive treatment assignment in experiments for policy choice." *Econometrica* 89, no. 1 (2021): 113-132.
- Lai, Tze Leung, and Herbert Robbins. "Asymptotically efficient adaptive allocation rules." *Advances in applied mathematics* 6, no. 1 (1985): 4-22.