

1 Brain Teasers

- **Information Encoding:** The type of problems ask to represent & discern different states of a system using numbers/bits. k^n states can be represented by n k -ary digits: say, 00, 01, 10, 11 for 4 states.

- **1000 Wine Bottles:** The system has 1000 states, live/die of one tester is a bit of information. So we need $\lceil \log_2 N \rceil$. Eaten-sion with k poison bottles is suprisingly much more complex.
- **12 Balls and a Balance:** Find the defect ball (heavier or lighter) with 3 measurements. *Analysis: (1) For single measurement of a heavier ball, 3-way operation $\{L, R, \emptyset\}$ corresponds to the 3 outcome states. (2) 3^3 outcomes to discern 12×2 results.*

Operation on Ball i in 3 measurements	L	R		L	R	L	R	L	R	L	R	L	R
Ball i is heavier	1	2	3	4	5	6	7	8	9	10	11	12	

The operations read: 1, 4, 7, 11 | 6, 9, 8, 10, ... Flip the outcome sign to map to the 12 lighter ball cases.

- **Weighing Coins I:** 5 bags each has 100 coins (10g), except 1 bag has fake coins (11g). Find it with 1 weighing on digital scale. The system has 5 states, so the outcome should take at least 5 different values. Take i coins from the i -th bag, total weight = 15 coins \times 10 grams + i grams = 150 + i .
- **Weighing Coins II:** Each bag may have fake coins (11g). Follow-up: fake coins can be 9 or 11g. There are 2^5 states. Draw 2^{k-1} coins from the k -th bag. Allowing -1 as an extra individual state, the system now has 3^5 states. Draw 3^{k-1} coins from the k -th bag.
- **Identify Polynomial:** Function $p(x)$ computes value of a polynomial of unknown order and non-negative integer coefficients wrt. x , identify the polynomial coefficients with minimal calls of p . First time: $z = p(1) = \sum_{j=0}^n a_j$, we'll know all $0 < a_j \leq z$ and the order $n \leq z-1$. Second time: $M = p(z+1) = \sum_{j=0}^{z-1} a_j(z+1)^j$. Do a base $(z+1)$ expansion of number M , the results are $\{a_j\}$'s.
- **Hat Colors I:** N people wearing R/B hats, called sequentially to declare the color of his hat and everyone else can hear it. Maximize # of correct declarations. Suppose the first person sees (N_R, N_B) . He encodes $C_1 = N_B \% 2$ with R/B and says it. Anyone k has own code $C_k = N_B \% 2$ or $(N_B - 1) \% 2$. So k in red if $C_k = C_1$, else blue.
- **Hat Colors II:** There are m different colors. Encode $C_1 = \{\sum_{i=1}^m (i-1)N_i\} \% m = S_1 \% m$ with m -colors. Anyone wearing color j sees $N_j - 1$ in that color, hence the code $C_k = \{\sum_{i \neq j} (i-1)N_i + j(N_j - 1)\} \% m = (S_1 - j) \% m, j \in [0, m-1]$.

- **Pigeonhole Principle:** $mn + 1$ items to put in n containers, at least 1 has $m + 1$ items.
 - **Handshake:** 10 ppl and I at the party, each shook hands with me, then at least 2 shook same number of hands. Each shook 1-25 hands (holes), and there are 26 ppl (pigeons).
 - **Friend Circle:** there are 6 people at the party, either at least 3 are friends (with each other), or at least 3 are strangers. There are 1-6 friend circles. (3-6): 3 strangers by selecting 1 from each circ. (1-2): 6 pigeons in 1/2 holes, at least 3 friends..
- **Summation:** $\sum_{k=1}^n k^2 = n(n+1)(2n+1)/6$. $\sum_{k=0}^{n-1} a_0 q^k = a_0 \frac{1-q^n}{1-q}$
 - **Missing 2 Integers:** Sum k and k^2 , solve 2 equations.

- **Two Eggs:** Given 2 eggs, find the n -th floor where the eggs starts to break. Minimize worst case #throws. Assume x throws in total. Drop E1 at x -th floor; if it breaks, try E2 from 1, 2, ..., $(x-1)$; if not, Drop B1 at $x + (x-1)$ since only $x-2$ drops left for E2. Worst case: $x + (x-1) + \dots + 2 + 1 \geq 100$. Follow ups - 100 eggs: binary search. k eggs: see the DP section.

• Multi-Agenet Induction:

- **Flip Glass:** Green Book pp.17. n men get summoned randomly, can flip the Glass (initially 0) or don't. At some point, a man can correctly identify everyone has been summoned at least once. The man who make the call ("Counter"): $1 \rightarrow 0, 0 \rightarrow 0$. Every-one else: $0 \rightarrow 1, 1 \rightarrow 1$, do nothing if he has flipped once.

• Mathematical Induction:

- **Coin Split:** Recursively split n coins into 2 piles until all piles has 1 coin. Find $S_n = \sum_{i \in \{\text{all splits}\}} x_{i1} x_{i2}$. Claim: $S_n = S_{n-1} + n - 1 = n(n-1)/2, \forall n = 2, \dots, N$. Consider $\forall k = 1, \dots, N$: $S_{N+1} = S_k + S_{N+1-k} + k(N+1-k) = k(k-1)/2 + (N+1-k)(N-k)/2 + k(N+1-k) = N(N+1)/2$.
- **Chocolate Bars:** Number of splits needed to break $m \times n$ chocolate bar into 1×1 pieces. Hack: number of pieces $\neq 1$ for each break, so $1 + S_{mn} = mn$. Induction Claim: $S_{mn} = mn - 1, \forall m \leq M, n \leq N$, show for $S_{M+1, N}$ and $S_{M, N+1}$
- **Kelly Criterion:** Two-outcomes bet with a p probability to win: (1) lose all bet, (2) win a fixed proportion (payoff odds, α) of the bet. Optimal strategy is to bet a fixed fraction of current wealth, f . $f^* = \operatorname{argmax} (1 + \alpha f)^p (1 - f)^{1-p}$, take log. $f^* = (\alpha p + p - 1)/\alpha$. General case: $f^* = \operatorname{argmax} (1 + \alpha f)^p (1 - \beta f)^q, f^* = p/\beta - q/\alpha$.

• Miscellaneous:

- **Horse Race:** Find 3 fastest horses out of 25 with fewest races, each race can have at most 5 horses. Let the following matrix be the ordering of the horses:

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25

 First run 5 races row-wise, then run the 6-th race along first column. Now 1 has to be 1st; 2,6 are possible for 2nd-place; 3,7,11 for 3rd. 7-th race among 2,6,3,7,11.
- **Hat Colors III:** 7 ppl randomly assigned 7 colors (can be duplicates), guess simultaneously, s.t. at least one correct guess.

2 Calculus

• Convergent Series & Tests:

- **Integral Test:** The *Harmonic series* diverges: Sum of rectangles $\sum_{i=1}^n \frac{1}{k} > \int_1^{n+1} \frac{1}{x} dx = \log(n+1) \rightarrow \infty$.
- **Comparison Test:** $\sum_{k=1}^n \frac{1}{k^2}$ convergent. $\frac{1}{k^2} < \frac{1}{k(k-1)}$. Hence $\sum_{k=2}^n \frac{1}{k} < \sum_{k=2}^n 2(\frac{1}{k-1} - \frac{1}{k}) = 1 - \frac{1}{n} \rightarrow 1$.

3 Linear Algebra

3.1 Fundamental Subspace, Det, Invertibility, etc

- For $n \times n$ square matrix A , the following statements are equivalent (TFAE), i.e. they are either all True or all False for any given A :
 - A is invertible, $Ax = b$ has exactly one solution in \mathbb{R}^n .
 - A has full rank, the columns of A are linearly independent.

- $\operatorname{Ker}(A)$ is trivial, i.e. $Ax = 0$ has only solution $x = 0$.
- $\det(A) \neq 0$, A doesn't have 0 eigenvalue.
- $\operatorname{span}\{\operatorname{Col}(A)\} = \mathbb{R}^n$.

• **Trace:** Sum of eigenvalues, *Cyclic property:* $\operatorname{tr}(ABC) = \operatorname{tr}(BCA) = \operatorname{tr}(CAB) \neq \operatorname{tr}(ACB)$.

3.2 Eigenvalues and Eigenvectors

Square matrix A : $Av = \lambda v$. λ is eigval of $A \iff$ it's a root of the characteristic polynomial $P_A(t) = \det(tI - A)$, i.e. λ solves $\det(\lambda I - A) = 0$

• Some Puzzles

- **Eigvals of Self-Outerproduct:** $A = vv^\top$. Notice that: $Av = vv^\top v = (v^\top v)v$. So $v^\top v$ is eigval that corresponds to eigvec v . Also: $\operatorname{tr}(vv^\top) = \operatorname{tr}(v^\top v) = v^\top v$ which is sum of all eigvals $\Rightarrow 0$ is eigval with multiplicity $n-1$, none zero $v^\top v$ has multiplicity 1.

• **Pseudoinverse:** $X \in \mathbb{R}^{m \times n}$ has full rank, $X^+ = (X^\top X)^{-1}X$ if X has linearly indep. cols; $X^+ = X^\top (XX^\top)^{-1}$ lindep. rows.

• SVD:

• **Covariance Matrix:** $\operatorname{Cov}[X] = \mathbb{E}[(X - \mathbb{E}[X])(X - \mathbb{E}[X])^\top]$.

4 Optimization

• KKT Conditions:

• **Mean-Variance Problem:** $\max_x \mu^\top x - \frac{\gamma}{2} x^\top \Sigma x$, s.t. $1^\top x = 1$.

$$x^* = \frac{\lambda}{1^\top \Sigma^{-1} \mu} \Sigma^{-1} \mu + \frac{1-\lambda}{1^\top \Sigma^{-1} 1} \Sigma^{-1} 1; \text{ where } \lambda = \frac{1^\top \Sigma^{-1} \mu}{\gamma}$$

5 Stochastic Models

5.1 Puzzles

• Mutually Exclusive, Collectively Exhaustive Events

- A flips $n+1$ coins, B flips n coins. $\mathbb{P}(A \text{ has more heads})?$ A flips the first n coins. $\{A \text{ more heads}\}, \{A \text{ less heads}\}, \{Equal\}$ MECE. $2p_1 + p_2 = 1$. $\mathbb{P}(E) = p_1 + p_2/2 = 1/2$.

- 52 cards, draw 2 without replacement. $\mathbb{P}(1st > 2nd)?$

$$2p + p_{tie} = 1. p_{tie} = \frac{13C1 \times 4C2}{52C2} = 1/17. p = 8/17.$$

- **Taking Seat on Plane:** 100 Seats. Person #1 is drunk.

$\mathbb{P}(\text{Person \#100 gets his seat})$.

Say k 's seat is occupied, equal prob to take seat #1 or #100. If k takes #1, all after k gets own seat; if #100, the last person will have to take seat #1. $\mathbb{P} = 1/2$.

- **Russian Roulette** (Green Book pp.81) Single bullet in the gun, be the first or second? $\mathbb{P}(D|\text{First}) = \mathbb{P}(\text{bullet in 1,3,5 slot}) = 1/2$.

- Spin the revolver after each trial: $\mathbb{P}(D|\text{First}) = \frac{1}{6} + (\frac{5}{6})^2 \frac{1}{6} + (\frac{5}{6})^4 \frac{1}{6} + \dots = \lim_{n \rightarrow \infty} \frac{1}{6} \frac{1 - (25/36)^n}{1 - 25/36} = \frac{6}{11}$. So we want to be the second player. Recursion solution: $\mathbb{P}(D|\text{First}) = \frac{1}{6} + (\frac{5}{6})^2 \mathbb{P}(D|\text{First})$.
- There are 2 bullets in the gun, I'm the second player. Opponent survived the first shot, should I spin or not? Obviously, $\mathbb{P}(D|\text{spin}) = \frac{1}{3}$. $\mathbb{P}(D|\text{not spin}) = \frac{\binom{4}{1} \binom{1}{1}}{\binom{5}{2}} = \frac{2}{5}$. Spin!

- There are 2 consecutive bullets in the gun: $\mathbb{P}(D|\text{spin}) = \frac{1}{3}$ regardless of the positioning of the bullet. But $\mathbb{P}(D|\text{not spin}) = \frac{\{\text{Bullet at 2,3}\}}{\{\text{Consecutive bullets in 2,3,4,5,6}\}} = \frac{1}{4}$. Don't spin!

• Compute Probabilities by Conditioning.

- **Extinction Problem:** A creature dies/stays same/spawn to 2/spawn to 3 with equal prob. $E = \{\text{Extinction}\}$. $\mathbb{P}(E) = \sum_{i=0}^3 \mathbb{P}(E|F_i) \mathbb{P}(F_i)$. Turns out that $\mathbb{P}(E|F_i)$ is equivalent to playing this game independently i -times (for i offsprings), hence $\mathbb{P}(E|F_i) = \mathbb{P}(E)^i$. $\Rightarrow \mathbb{P}(E) = \frac{1}{4}(1 + \mathbb{P}(E) + \mathbb{P}(E)^2 + \mathbb{P}(E)^3)$.

• Compute Expectation by Linearity.

- **Coupon Collection I.** Equal probability to see any 1 of n different types of coupons, expected number of trials until seen all n types. Let X_k be number of trials to open to see a new type when $(k-1)$ types have already been collected $X_k \sim \text{Geometric}(\frac{n-(k-1)}{n})$.

$$\mathbb{E}[X] = \sum_{k=1}^n X_k = \sum_{k=1}^n \frac{n}{n-(k-1)} = 1 + \frac{n}{n-1} + \frac{n}{n-2} + \dots + n$$

• Compute Expectation by Markov Chain.

- Coupon Collection II.

• Conditional Expectations: Properties:

1. **Pulling out independent factors:** If $X \perp \mathcal{F}$, then $\mathbb{E}[X|\mathcal{F}] = \mathbb{E}[X]$. If $X \perp \sigma(Y, \mathcal{F})$, then $\mathbb{E}[XY|\mathcal{F}] = \mathbb{E}[X] \mathbb{E}[Y|\mathcal{F}]$, note that both independence to Y and \mathcal{F} are required.
2. **Pulling out known factors:** If $X \in m\mathcal{F}$, then $\mathbb{E}[XY|\mathcal{F}] = X \mathbb{E}[Y|\mathcal{F}]$; For RV Y, Z : $\mathbb{E}[f(Z)Y|\sigma(Z)] = f(Z) \mathbb{E}[Y|\sigma(Z)]$

- **Expectation of OLS beta:** Sample (x_i, y_i) uniformly from region: $\{0 \leq x, y \leq 1; x + y \geq 1.5\}$, and run OLS with intercept between sample $\mathbf{y} = \beta_0 + \beta_1 \mathbf{x}$. Compute $\mathbb{E}[\hat{\beta}_0], \mathbb{E}[\hat{\beta}_1]$. Let $\mathbf{X} = (\mathbf{u}, \mathbf{x})$

$$\mathbb{E}[\hat{\beta}|\mathbf{X}] = \mathbb{E}[(\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}|\mathbf{X}] = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbb{E}[\mathbf{y}|\mathbf{X}]$$

$$\text{Turns out: } y_i|x_i = \frac{1}{2}(1 + (\frac{3}{2} - x_i)) = \frac{5}{4} - \frac{1}{2}x_i. \mathbb{E}[\mathbf{y}|\mathbf{X}] = \mathbf{X}(\frac{5}{4}, -\frac{1}{2})^\top. \mathbf{X}^\top \mathbf{X} \text{ cancels out, } \mathbb{E}[\hat{\beta}_0, \hat{\beta}_1] = \frac{5}{4}, -\frac{1}{2}.$$

5.2 Measure space theory

• Inequalities: Pass

• Convergence of RVs:

- **Convergence In Prob / Almost Surely:** $X_n \xrightarrow{a.s.} X \iff \mathbb{P}(\lim_{n \rightarrow \infty} X_n = X) = 1$: the limit of $X_n(s)$ is X with measure 1 wrt $s \in \Omega$, an analogy of pointwise convergence in the measure space (SLLN). $X_n \xrightarrow{p} X \iff \lim_{n \rightarrow \infty} \mathbb{P}(|X_n - X| > \epsilon) = 0, \forall \epsilon > 0$: the probability of unusual outcome becomes smaller and smaller (Consistent estimator, WLLN).
- **Convergence In Prob but not A.S.:** $X_1(\omega) = \mathbb{1}(w \in [0, \frac{1}{2}]), X_2(\omega) = \mathbb{1}(w \in [\frac{1}{2}, 1]), X_3(\omega) = \mathbb{1}(w \in [0, \frac{1}{3}]), X_4(\omega) = \mathbb{1}(w \in [\frac{1}{3}, \frac{2}{3}]), \dots, \omega \sim \text{Uniform}[0, 1]$.

5.3 Named Distributions

• Gaussian:

- **Partition:** $\mathbf{x} = (x_1; x_2) \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}), \boldsymbol{\mu} = (\boldsymbol{\mu}_1; \boldsymbol{\mu}_2), \boldsymbol{\Sigma} = (\boldsymbol{\Sigma}_{11}, \boldsymbol{\Sigma}_{12}; \boldsymbol{\Sigma}_{21}, \boldsymbol{\Sigma}_{22})$. Then $\mathbf{x}_1|x_2 \sim \mathcal{N}(\boldsymbol{\mu}_{x_1|x_2}, \boldsymbol{\Sigma}_{x_1|x_2})$, where $\boldsymbol{\mu}_{x_1|x_2} = \boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}(\mathbf{x}_2 - \boldsymbol{\mu}_2); \boldsymbol{\Sigma}_{x_1|x_2} = \boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21}$. Marginal $\mathbf{x}_1 \sim \mathcal{N}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_{11})$.

- **Bayes Theorem:** Given $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Gamma}), \mathbf{y}|\mathbf{x} \sim \mathcal{N}(\mathbf{A}\mathbf{x} + \mathbf{b}, \boldsymbol{\Omega})$. Then: $\mathbf{y} \sim \mathcal{N}(\mathbf{A}\boldsymbol{\mu} + \mathbf{b}, \boldsymbol{\Omega} + \mathbf{A}^\top \boldsymbol{\Gamma} \mathbf{A})$
 $\mathbf{x}|\mathbf{y} \sim \mathcal{N}(\boldsymbol{\Sigma}[\mathbf{A}^\top \boldsymbol{\Omega}^{-1}(\mathbf{y} - \mathbf{b}) + \boldsymbol{\Gamma}^{-1}\boldsymbol{\mu}], \boldsymbol{\Sigma}), \boldsymbol{\Sigma} = (\boldsymbol{\Gamma}^{-1} + \mathbf{A}^\top \boldsymbol{\Omega}^{-1} \mathbf{A})^{-1}$.

• Chi-Squared:

- **Quad Form:** If m -vector $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$, then $\mathbf{x}^\top \boldsymbol{\Sigma}^{-1} \mathbf{x} \sim \chi^2(m)$.
- **Projection:** n -vector $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, \mathbf{P} is rank- r projection matrix onto a subspace of E^n , then $\mathbf{z}^\top \mathbf{P} \mathbf{z} \sim \chi^2(r)$.

• ss

6 OLS

• OLS Assumptions And Their Purposes:

- **Linearity:** Correct specification of the linear functional form.
- **No Perfect Multicollinearity:** Assures the min-square problem has a unique solution (the linear system: FOC) - the $\hat{\beta}$ are identifiable.
- **Exogeneity of X:** $\mathbb{E}[\mathbf{u}|\mathbf{X}] = \mathbf{0}$. This allows us to derive $\mathbb{E}[\hat{\beta} - \beta] = \mathbb{E}[(\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{u}] = \mathbf{0}$ (tower property), i.e. OLS estimator is *consistent*. **Relaxation:** Instrumental Variables.
- **X has Finite Second Moment:** This means $\frac{1}{n} \mathbf{X}^\top \mathbf{X} \xrightarrow{p} \mathbf{S}_{\mathbf{X}^\top \mathbf{X}}$, i.e. converge in probability to a finite and positive semi-definite matrix $\mathbf{S}_{\mathbf{X}^\top \mathbf{X}}$. This is a much-used condition in deriving asymptotic properties, including all of the asymptotic tests and consistency.
- **Error Term Distribution:** There are several versions, from strongest to weakest:
 1. **Normality:** $\mathbf{u} \sim \text{i.i.d. } \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I})$. \Rightarrow Finite sample statistical (exact) tests; and makes OLS equivalent to MLE.
 2. **Spherical Errors:** \mathbf{u} follows some i.i.d. (not normal) distribution with equal variance. This allows asymptotic statistical tests, by applying CLT.
 3. **No Serial Correlation:** $\text{Cov}[\mathbf{u}|\mathbf{X}] = \text{diag}\{\sigma_k^2\}_1^n$. This corresponds to *Weighted Least Squares* and other Heteroskedasticity-Robust method.
 4. **(General) Deterministic Cov Matrix:** $\text{Cov}[\mathbf{u}|\mathbf{X}] = \boldsymbol{\Omega}$, corresponds to *Generalized Least Squares*.

- **FWL Theorem:** $\mathbf{y} = \mathbf{X}_1 \beta_1 + \mathbf{X}_2 \beta_2 + \mathbf{u}$, $\mathbf{M}_1 \mathbf{y} = \mathbf{M}_1 \mathbf{X}_2 \beta_2 + \mathbf{u}$ has same ols estimator and residuals. $\hat{\beta}_2 = (\mathbf{X}_2^\top \mathbf{M}_1 \mathbf{X}_2)^{-1} \mathbf{X}_2^\top \mathbf{M}_1 \mathbf{y}$.

$$\mathbf{M}_1 \mathbf{y} = \mathbf{M}_1 \mathbf{X}_2 \hat{\beta}_2 + \mathbf{M}_1 \mathbf{M}_X \mathbf{y} \Rightarrow \mathbf{M}_1 \mathbf{y} - \mathbf{M}_1 \mathbf{X}_2 \hat{\beta}_2 = \mathbf{M}_X \mathbf{y}.$$

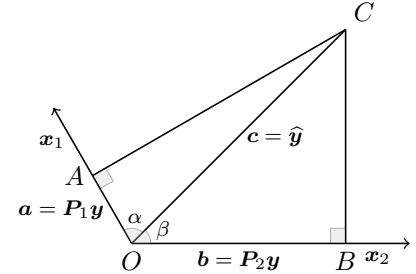
- Adding \mathbf{X}_2 into $\mathbf{y} = \mathbf{X}_1 \beta_1 + \mathbf{u}$ is likely to cause change in $\hat{\beta}_1$, won't if $\mathbf{M}_X \mathbf{X}_1 = \mathbf{X}_1$ i.e. $\mathbf{X}_1 \perp \mathbf{X}_2$.

- **R Squared:** Let $\boldsymbol{\iota}$ be column of n "1"s. Projecting \mathbf{y} onto $\boldsymbol{\iota}$ yields a column of n copies of the mean of \mathbf{y} . Easy to verify that \mathbf{H}_ι has $(n \times n)$ entries of $\frac{1}{n}$ s.

$$R^2 = 1 - \frac{\text{SSRes}}{\text{SSTot}} = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} = \frac{\mathbf{y}^\top (\mathbf{M}_\iota - \mathbf{M}_X) \mathbf{y}}{\mathbf{y}^\top \mathbf{M}_\iota \mathbf{y}}$$

- When \mathbf{X} has intercept term, $R^2 \geq 0$ for sure, because $(\mathbf{M}_\iota - \mathbf{M}_X)$ is symmetric-idempotent, thus positive-semidefinite. If no intercept, could be negative: extreme counter example: $y_i = 100 + \text{small } \epsilon_i, x_i = 1, 2, \dots$
- $\mathbf{y} \sim \mathbf{x}_1$ yields R_1^2 , $\mathbf{y} \sim \mathbf{x}_2$ yields R_2^2 , what's the range of R^2 for $\mathbf{y} \sim \mathbf{x}_1 + \mathbf{x}_2$ (WLOG assume stuffs have mean 0).

The answer is $R^2 \in [\max\{R_1^2, R_2^2\}, 1]$. R^2 can actually be derived as a function of $\text{Corr}(\mathbf{x}_1, \mathbf{y}) = R_1, \text{Corr}(\mathbf{x}_2, \mathbf{y}) = R_2$, and $\text{Corr}(\mathbf{x}_1, \mathbf{x}_2) =: \rho_{x_1, x_2}$ by the following geometric proof:



$$\begin{aligned} \rho_{x_1 x_2} &= \cos(\alpha + \beta) = \cos \alpha \cos \beta - \sin \alpha \sin \beta \\ &= \frac{ab}{c^2} - \sqrt{1 - \frac{a^2}{c^2}} \sqrt{1 - \frac{b^2}{c^2}} \\ \Rightarrow R^2 &= \frac{c^2}{y^2} = \frac{(a^2 + b^2 - 2ab\rho_{x_1 x_2})/y^2}{1 - \rho_{x_1 x_2}^2} = \frac{R_1^2 + R_2^2 - 2R_1 R_2 \rho_{x_1 x_2}}{1 - \rho_{x_1 x_2}^2} \end{aligned}$$

Obvious that R^2 can be 1 when $\mathbf{y} \in \text{span}\{\mathbf{x}_1, \mathbf{x}_2\}$. $R^2 - R_1^2 \geq 0$ because that reduces to $\rho_{x_1 x_2}^2 R_1^2 + R_2^2 - 2\rho_{x_1 x_2} R_1 R_2 = (\rho_{x_1 x_2} R_1 - R_2)^2$.

• Misc Algebra Questions:

- **Product of Projection Matrices:** If $\text{span}\{\mathbf{X}_1\} \subseteq \text{span}\{\mathbf{X}\}$, then $\mathbf{P}_X \mathbf{P}_{X_1} = \mathbf{P}_{X_1}$ (proj onto the smaller subspace); and $\mathbf{M}_X \mathbf{M}_{X_1} = \mathbf{M}_X$ (\perp proj onto the bigger subspace).
- **Zero x2 Coefficient:** $\mathbf{P}_1 \mathbf{y} = \hat{\beta}_1 \mathbf{x}_1, \mathbf{P}_X \mathbf{y} = \hat{\beta}_1' \mathbf{x}_1 + 0 \mathbf{x}_2$. Pre-multiply both ols with \mathbf{P}_1 , since $\mathbf{P}_1 \mathbf{P}_X = \mathbf{P}_1$ (always, when $\mathbf{x}_1 \in \text{span}\{\mathbf{X}\} \Rightarrow \hat{\beta}_1 = \hat{\beta}_1'$).
- **Exchange x, y:** $\hat{\beta}_{y \sim x} \hat{\beta}_{x \sim y} = (\mathbf{x}^\top \mathbf{y})^2 / (\mathbf{x}^\top \mathbf{x} \cdot \mathbf{y}^\top \mathbf{y}) = \text{Corr}^2(\mathbf{x}, \mathbf{y}) \leq 1$. I.e. $\hat{\beta}_{y \sim x} \leq 1/\hat{\beta}_{x \sim y}$.
- **Duplicate Data:** $\text{Var}[\hat{\beta}] = \frac{\sigma^2}{2} (\mathbf{X}^\top \mathbf{X})^{-1}$, underestimated, half of the correct value. $R^2, \hat{\beta}$ doesn't change. Implication is that correlation in \mathbf{X} downplays the variance estimate.
- **beta and Correlation:** For simple linear regression with intercept or without intercept while the RV X, Y has 0 mean: $\hat{\beta} = \text{Corr}[X, Y]$. Application: For zero mean RVs X_1, X_2, Y , suppose $\rho_{X_1, Y} = 0.1, \rho_{X_2, Y} = 0.2$, run $\mathbf{y} \sim \mathbf{x}_1 + \mathbf{x}_2$, will $\hat{\beta}_2$ be negative?

• Prediction Error:

- **Leverage:** $(\hat{\beta}^{(t)} - \beta)$. Removing t -th obs = Adding unit basis \mathbf{e}_t into the regression. $\mathbf{y} = \mathbf{X} \beta' + \alpha \mathbf{e}_t + \mathbf{u}, \mathbf{M}_t \mathbf{y} = \mathbf{M}_t \mathbf{X} \beta^{(t)} + \mathbf{u}$. $\mathbf{X} \hat{\beta} = \mathbf{H}_X \mathbf{H}_X \mathbf{e}_t, \mathbf{y} = \mathbf{H}_X (\mathbf{X} \hat{\beta}^{(t)} + \hat{\alpha} \mathbf{e}_t)$. Hence $(\hat{\beta}^{(t)} - \hat{\beta}) = -\hat{\alpha} (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{e}_t = \frac{-\hat{\alpha}_t}{1 - h_{tt}} (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{x}_t^\top$. ($\hat{\alpha}$ obtained by another FWL: $\mathbf{M}_X \mathbf{y} = \alpha \mathbf{M}_X \mathbf{e}_t + \mathbf{u}$, and $h_{tt} = \mathbf{e}_t^\top \mathbf{H}_X \mathbf{e}_t = \cos^2 \langle \mathbf{e}_t, \mathbf{H}_X \mathbf{e}_t \rangle$)

• Hypothesis Testing:

- **T-Test:** $\mathbf{y} = \mathbf{X}_1 \beta_1 + \mathbf{x}_2 \beta_2 + \mathbf{u}$. $\mathbf{z}(\hat{\beta}_2) = \hat{\beta}_2 / \text{se}(\hat{\beta}_2) = \mathbf{x}_2^\top \mathbf{M}_1 \mathbf{y} / \sigma(\mathbf{x}_2^\top \mathbf{M}_1 \mathbf{x}_2)^{1/2} \sim \mathcal{N}(0, 1)$ under the null hypothesis $\beta_2 = 0$; $\text{RSS}/\sigma^2 = \mathbf{y}^\top \mathbf{M}_X \mathbf{y} / \sigma^2 \sim \chi^2(n - p)$, by the projection

property of χ^2 . The t-statistic for β_2 :

$$t(\hat{\beta}_2) = \frac{z(\hat{\beta}_2)}{\sqrt{\chi^2_{n-p}/(n-p)}} = \frac{\mathbf{x}_2^\top \mathbf{M}_1 \mathbf{y}}{\sigma(\mathbf{x}_2^\top \mathbf{M}_1 \mathbf{x}_2)^{\frac{1}{2}}} \left(\frac{RSS}{\sigma^2(n-p)} \right)^{-\frac{1}{2}}$$

- F-Test:

- Regression Toward the Mean (Mediocrity):** The son of a tall father is on average shorter than the father - closer to the human mean height. Consider the bivariate normal partition: $\mathbb{E}[X_2|X_1] - \mu_2 = \rho\sigma_2\sigma_1^{-1}(X_1 - \mu_1) \Rightarrow \frac{\mathbb{E}[X_2|X_1] - \mu_2}{\sigma_2} = \rho \frac{X_1 - \mu_1}{\sigma_1}$, i.e. the average X_2 given X_1 is closer to X_2 -mean than X_1 to X_1 -mean, with $|\rho| < 1$. If X_1, X_2 has *identical marginal dist*: $\mu \leq \mathbb{E}[X_2|X_1 = x_1] < x_1$ for observation $x_1 > \mu$ (*the tall father*).

7 Time Series

- p-th Order Difference Equation:** $y_t = \phi_1 y_{t-1} + \phi_2 y_{t-2} + \dots + \phi_p y_{t-p} + w_t$. In vector form: $\boldsymbol{\xi}_t = \mathbf{F} \boldsymbol{\xi}_{t-1} + \mathbf{v}_t$, where

$$\boldsymbol{\xi}_t = \begin{pmatrix} y_t \\ y_{t-1} \\ \vdots \\ y_{t-(p-1)} \end{pmatrix}_{p \times 1} \quad \mathbf{F} = \begin{pmatrix} \phi_1 & \phi_2 & \phi_3 & \dots & \phi_{p-1} & \phi_p \\ 1 & 0 & 0 & \dots & 0 & 0 \\ 0 & 1 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 & 0 \end{pmatrix}_{p \times p}$$

- LAG Operators:** For $|\phi| < 1$, $|\theta| < 1$, y_t bounded:
 $\lim_{t \rightarrow \infty} (1 - \phi L)(1 + \sum_{i=1}^t \phi^i L^i) y_t = \lim_{t \rightarrow \infty} (y_t - \phi^{t+1} y_{-1}) \rightarrow y_t$
 $\lim_{t \rightarrow \infty} (1 + \theta L)(1 + \sum_{i=1}^t (-\theta)^i L^i) y_t = \lim_{t \rightarrow \infty} (y_t + (-\theta^t) \theta y_{-1})$
- Lag Polynomial:** Factor the order- p polynomial: $(1 - \sum_{i=1}^p \phi_i L^i) = \prod_{i=1}^p (1 - \lambda_i L^i) = L^p \prod_{i=1}^p (L^{-1} - \lambda_i) = \lambda^{-p} (\lambda^p - \sum_{i=1}^p \phi_i \lambda^{p-i})$ is therefore equivalent to finding roots of $\lambda^p - \phi_1 \lambda^{p-1} - \dots - \phi_{p-1} \lambda - \phi_p = 0$, which are the *eigenvalues of matrix F*. The limit equation holds if $|\lambda_i| < 1$ for all roots (*lie within the unit circle for $\lambda_i \in \mathbb{C}$*).
- Outer Product (Leading-p Autocov Matrix):** Suppose $\boldsymbol{\xi}_{t-j} \mathbf{v}_t^\top = 0$ (*analogy of $\mathbb{E}[Y_{t-j} \epsilon_t] = 0, \forall j > 0$*), then $\boldsymbol{\xi}_t \boldsymbol{\xi}_t^\top = \mathbf{F} \boldsymbol{\xi}_{t-1} \boldsymbol{\xi}_{t-1}^\top \mathbf{F}^\top + \mathbf{v}_t \mathbf{v}_t^\top \Rightarrow \boldsymbol{\Sigma} = \mathbf{F} \boldsymbol{\Sigma} \mathbf{F}^\top + \sigma^2 \mathbf{I}_p$ for autocov. *It turns out:* $\mathbf{I}_{p^2} \text{vec}(\boldsymbol{\Sigma}) = (\mathbf{F} \otimes \mathbf{F}) \text{vec}(\boldsymbol{\Sigma}) + \text{vec}(\sigma^2 \mathbf{I}_p)$. Autocov is given by the first p -entries of $\sigma^2 (\mathbf{I}_{p^2} - \mathbf{F} \otimes \mathbf{F})^{-1}$.

- Stationary Processes:** Y_t is *mean stationary* if $\mathbb{E}[Y_t]$ is constant. Y_t is *covariance stationary* if (1) $\mathbb{E}[Y_t] = \mu$, (2) $\mathbb{E}[(Y_t - \mu)(Y_{t-j} - \mu)] = \gamma_j \forall t, j$: *first and second moments don't depend on t*. Y_t is *strict stationary* if joint distribution $F_{Y_t, Y_{t+j_1}, \dots, Y_{t+j_n}} = F(j_1, \dots, j_n)$, *doesn't depend on t*. Linear combination of stationary process is stationary.

- Ergodicity:** Means time average (just 1 sample path) converges to state space average, i.e. expectation. For cov-stationary Y_t : $\sum_{j=0}^{\infty} |\gamma_j| \leq \infty \Rightarrow \frac{1}{T} \sum_{t=1}^T Y_t^{(1)} \xrightarrow{P} \mathbb{E}[Y_t]$. For Gaussian process Y_t that's sufficient for ergodicity for all moments.
- MA(q):** $Y_t = \mu + \epsilon_t + \sum_{k=1}^q \theta_k \epsilon_{t-k} = \mu + (1 + \sum_{i=1}^q \theta_i L^i) \epsilon_t$. *Is always stationary regardless of coefficients.*

$$\mathbb{E}[Y_t] = \mu, \quad \gamma_h = \begin{cases} \sigma^2 \sum_{s=h}^q \theta_{s-h} \theta_s & h \leq q \\ 0 & h > q \end{cases}$$

- AR(1):** $Y_t = c + \phi_1 Y_{t-1} + \epsilon_t \iff (1 - \phi_1 L) Y_t = c + \epsilon_t$. Stationary when $|\phi| < 1$: $\mathbb{E}[Y_t] = c(1 - \phi L)^{-1} = c/(1 - \phi)$, $\gamma_h = \sigma^2 \sum_{s=0}^{\infty} \phi^s \phi^{s+h} = \sigma^2 \phi^h / (1 - \phi^2)$.

- AR(p), Yule-Walker Equations:** Under the stationary condition, $Y_t = c + \sum_{i=1}^p \phi_i Y_{t-i} + \epsilon_t = \mu + \sum_{i=1}^p \phi_i (Y_{t-i} - \mu) + \epsilon_t$. Multiply by $(Y_{t-h} - \mu)$, note that Y_{t-j} contains ϵ_s up to $s = t - h$:

$$\gamma_h = \begin{cases} \phi_1 \gamma_{h-1} + \phi_2 \gamma_{h-2} + \dots + \phi_p \gamma_{h-p} & h = 1, 2, \dots (\infty) \\ \phi_1 \gamma_{-1} + \phi_2 \gamma_{-2} + \dots + \phi_p \gamma_{-p} + \sigma^2 & h = 0 \end{cases}$$

Note $\gamma_i = \gamma_{-i}$, the first p terms are given by (7), recursion of γ_h is also a p -th order difference equation. $\gamma_h = \sum_{i=1}^p \phi_i \gamma_{h-i}$, $h \geq 1$.

- ARMA(p, q):** $Y_t = c + \sum_{i=1}^p \phi_i Y_{t-i} + \epsilon_t + \sum_{j=1}^q \theta_j \epsilon_{t-j}$. *Stationary when all roots of $\lambda^p - \sum_{i=1}^p \phi_i \lambda^{p-i}$ are inside unit circle.* $\mathbb{E}[Y_t] = c/(1 - \phi_1 - \dots - \phi_p)$. Yule-Walker equation holds for $h > q$.
- ACF/PACF Plots:** $\xi_Y(k) = \text{Cov}[Y_t, Y_{t-k} | Y_{t-1}, \dots, Y_{t-k+1}]$ is the Pacf. The plot profile looks like:
 - AR(p):** ACF tails off (approaches 0 as $k \rightarrow \infty$); PACF cuts off at order p : autocorrelation passes on to arbitrary distant term, but through intermediate terms (goes away when conditioned on).
 - MA(q):** ACF cuts off at order q (no common ϵ terms); PACF tails off: when conditioned on middle terms, all $\{\epsilon\}$'s in between are correlated through linear constraints, thus passes on to any distant term.
 - ARMA(p, q):** ACF, PACF tails off. Mixture of both effects.

- Unit Root:** Let $\rho = \sum_{i=1}^p \phi_i$, $\zeta_i = \sum_{j=i+1}^p \phi_j$, can verify AR(p) operator equal to $\{(1 - \rho L) - (\zeta_1 L + \zeta_2 L + \dots + \zeta_{p-1} L^{p-1})(1 - L)\}$. If $\lambda^p - \sum_{i=1}^p \phi_i \lambda^{p-i} = 0$ has a unit root $\lambda = 1$: $\sum_{i=1}^p \phi_i = 1$.
 - ADF Test:** $\{(1 - \rho L) - (\zeta_1 L + \zeta_2 L + \dots + \zeta_{p-1} L^{p-1})(1 - L) Y_t\} = \epsilon_t \iff \Delta Y_t = (\rho - 1) Y_{t-1} + (\zeta_1 L + \zeta_2 L + \dots + \zeta_{p-1} L^{p-1}) \Delta Y_t + \epsilon_t$, where $\Delta Y_t = Y_t - Y_{t-1}$. $H_0: \rho - 1 = 0$ (unit root); $H_1: \rho - 1 < 0$ (stationary; explosive if $\rho - 1 > 0$). $t_{\rho-1}^{ols} \sim 1$ -tailed t under H_0 .
 - 1-D Random Walk:** $Y_t = Y_{t-1} + \epsilon$ is mean stationary. $\text{Var}[Y_t] = \sigma^2 t$. And has 1 unit root hence 1st-difference stationary process (Integrated order 1).

8 Machine Learning Misc

- L^2 (Ridge):
- Ridge Shrinkage:**
- PCA:**
- Misc**
- $\bar{X} \sim \mathcal{N}(\mathbb{E}[X], \sigma_X^2/n)$; $\hat{\sigma}_X = \sqrt{\sum (X_i - \bar{X})^2 / (n-1)}$; $\hat{s}_e = \hat{\sigma}_X / \sqrt{n}$.

10 Algorithm And Data Structures

10.1 Data Structures

- Linear Structures:** Array, List, Stack, Queue, etc..
 - Array Range Sum Problem:** Given an array, want to efficiently compute (many times) $\sum_{k=i}^j a_k$. The idea is to cache $\text{csum} = \text{np.cumsum}([0] + \text{arr})$, resulting in $O(1)$ $\text{RangeSum}(\text{arr}, i, j) = \text{csum}[j+1] - \text{csum}[i]$.
 - Max Size Subarray Sum = K:** Find the shortest subarray in arr that sums to K . Trick1: $S = \text{np.cumsum}(\text{arr})$. If $S_j - S_i = K$, then $\text{arr}[i+1, j]$ sums to K , array length is $j - i$. Trick2: Like the idea of **Two Sum**, use hash map to store $\{S_j : j\}$ pairs for each increment of j . For multiple j with same cumsum, use the minimum j as dict value since we are looking for maximum size.

```

1: function MAXSIZESUBARRAYSUMK(a, K)
2:   S, res ← 0           // Container for running cumsum & results
3:   prevSums ← dict()    // The "Two Sum" Trick
4:   for i in 0:len(a) do
5:     S ← S + a[i]
6:     if S == K then     // a[0:i] sums to K, must be max so far
7:       res ← i + 1
8:     else if S - K in prevSums then
9:       res ← max{res, i - prevSums[S - K]}
10:    if S not in prevSums then
11:      prevSums[S] = i
12:  return res           // Time complexity = O(n)

```

- Implement Que with 2 Stacks:** Amortized $O(1)$ time: calling n push/pop operations is $O(n)$. Idea is for each queue pop, move elements from 1st stack to 2nd, and then pop the 2nd stack.
- Implement Stack with Que:** pop-push all elements again when pushing new element. $O(n)$ time for single push (So stupid...).

- Heap & Priority Queue:** Max/Min-heap - parent node value \geq / \leq children. Insert (push) and Delete root (pop) $\sim O(\log n)$.

- Median in Data Stream:** Maintain maxheap and minheap for values below/above the median. Insertion is $O(\log n)$, computing running median is $O(1)$. *Caveat: Balance the size of 2 heaps.*
- Merge K sorted lists:** Use heap to find minimum among k lists, time improvement $O(nk) \rightarrow O(n \log k)$.
- Trap Rain Water II:** Suppose the shortest bar on the *closed boundary* of cell board K is $h_{\partial K} = \min_{i,j \in \partial K} h_{ij}$. Then any neighbor cell of $h_{\partial K}$ must trap $\max\{0, h_{\partial K} - h\}$ unit of rain.

```

1: function TRAPRAIN2(K)
2:   hq ← MinHeap([([K[i, j], i, j] For i, j ∈ ∂K]); S ← 0
3:   Mark ∂K as visited.
4:   while hq is non-empty do
5:     h_{∂K}, (i*, j*) ← hq.pop()
6:     for (i, j) in unvisited neighbors of K[i*, j*] do
7:       S ← S + max{0, h_{∂K} - h}
8:       hq.push(h, (i, j))
9:       Mark K[i, j] as visited.
10:  return S

```

10.2 2 Pointers & Sliding Window

- Trap Rain Water I:** $w_i = \min(\max\{h_{0:i}\}, \max\{h_{i:n}\}) - h_i$. Improvements: memoize the left/right highest pillar. $O(n^2) \rightarrow O(n)$ time, $O(n)$ space. Two pointers: $O(1)$ space.
- More Subarray Problems:**
 - Subarray Product Less Than K:** The idea is to maintain two pointers l, r , each time $r \rightarrow r + 1$ and product $\prod_{i=l}^{r+1} a_i < K$, we add all subarrays between $l, r + 1$ that ends at $r + 1$ to result, which is $(r + 1 - l)$. If the product $\geq K$, $l \rightarrow l + 1$.


```

1: function SUBARRAYPRODLESTHANK( $\mathbf{a}$ ,  $K$ )
2:   if  $K \leq 1$  then
3:     return 0
4:    $l, \Pi \leftarrow 0, 1$ ;  $res \leftarrow 0$ 
5:   for  $r$  in  $0:\text{len}(\mathbf{a})$  do                                // Increase right ptr
6:      $\Pi \leftarrow \Pi * a_r$ 
7:     while  $\Pi \geq K$  do                                     // Increase left ptr if  $\Pi$  is beyond  $K$ 
8:        $\Pi \leftarrow \Pi / a_l$ 
9:        $l \leftarrow l + 1$ 
10:     $res \leftarrow res + (r + 1 - l)$ 
11:   return  $res$  // Must stop when  $l = r = \text{len}(\mathbf{a})$ , time  $\sim O(2n)$ 

```

- **Min Size Subarray Sum $\geq K$:** Exactly the same template as last problem. Each time $r \rightarrow r+1$, examine all subarrays between $[l, r]$. Update running min with $(r-l+1)$ when $\sum_{j=l}^r a_j \geq K$, and increase l (Like last problem, there are only positive integers).

```

1: function MINSIZE SUBARRAYSUMGEQK( $\mathbf{a}$ ,  $K$ )
2:   if  $\text{len}(\mathbf{a}) == 0$  then
3:     return 0
4:    $l, \Sigma \leftarrow 0, 0$ ;  $res \leftarrow +\infty$ 
5:   for  $r$  in  $0:\text{len}(\mathbf{a})$  do                                // Increase right ptr
6:      $\Sigma \leftarrow \Sigma + a_r$ 
7:     while  $\Sigma \geq K$  do                                     // Increase left ptr if  $\Sigma$  is beyond  $K$ 
8:        $res \leftarrow \min\{res, r + 1 - l\}$  // Update "running min"
9:        $\Sigma \leftarrow \Sigma - a_l$ 
10:       $l \leftarrow l + 1$ 
11:   return  $res$  // Must stop when  $l = r = \text{len}(\mathbf{a})$ , time  $\sim O(2n)$ 

```

10.3 Divide-and-Conquer Recurrence

- **Master Theorem:** Recurrence relation $T(n) = aT(n/b) + f(n)$. $T(n)$ is the time for problem at scale n , base case $T(1) = \Theta(1)$. $f(n)$ is the extra time to split and combine the subproblems outside the recursion. Turns out: $T(n) = O(n^{\log_b a}) + \sum_{i=0}^{\log_b n} a^i f(n/b^i)$.
 - 1. $f(n) = O(n^{\log_b a - \epsilon})$. The leaf (base-case) problem dominates, there are $a^{\log_b n}$ of them. $T(n) = \Theta(n^{\log_b a})$.
 - 2. $f(n) = \Omega(n^{\log_b a + \epsilon})$, $af(n/b) \leq cf(n)$. Then the total splitting time dominates: $T(n) = \Theta(f(n))$.
 - 3. $f(n) = \Theta(n^{\log_b a} \log^k n)$, $k > 0$. The two parts equivalently dominate, $T(n) = \Theta(n^{\log_b a} \log^{k+1} n)$.

- **Quicksort:**
- **Quickselect:**

- Sub

10.4 Dynamic Programming

- **Fibonacci:** Recursion $O(2^n)$ time, $O(\log n)$ space. DP $O(n)$ time, $O(1)$ space by only caching $f(n-1), f(n-2)$.
- **Coin Change:** Let $f(n)$ be the min number of coins to sum up n .

$$f(n) = \begin{cases} 0 & n = 0 \\ +\infty \text{ (unable to sum up)} & n < 0 \\ \min_{c \in \text{coins}} \{f(n-c) + 1\} & \text{otherwise} \end{cases}$$

- **Subarray Problems:** The array constructed by some *contiguous* elements of array s . The number of subarrays of length n array is $1 + \dots + n = n(n+1)/2$.

- **Maximum Subarray Sum:** The key is to realize DP on $f(k)$: the maximum sum of subarray *ending* at k . $f(k) = \max\{f(k-1) + s_k, s_k\}$. solution is $\max_{1 \leq k \leq n} f(k)$.
- **Maximum Subarray Product:** Let $f(k)$ be the max product of subarray ending at k . Due to the existence of negative numbers, $f(k)$ can be evolved from $f(k-1)$, and also the *min* (most negative) product of subarray ending at k , let it be $g(k)$.

$$f(k) = \max\{a_k f(k-1), a_k g(k-1), a_k\}$$

$$g(k) = \min\{a_k f(k-1), a_k g(k-1), a_k\}$$

$O(1)$ space: keep track of maximum $f(k)$ so far.

- **Subsequence Problems:** A *subsequence* of s is generated by deleting some elements from s while *preserving relative order*. Often solved with a $O(n^2)$ DP. Entries $A_{i,j}$ of the DP table represent the solution to subproblem indexed by (i, j) , which can be: (1) $\mathcal{P}(x_{i:j}) \preceq \mathcal{P}(x)$; (2) $\mathcal{P}(x_{0:i}, y_{0:j}) \preceq \mathcal{P}(x, y)$; (3) $\mathcal{P}(x_{0:i}) \preceq \mathcal{P}(x)$. In (1),(2) state transition must be $O(1)$, while in (3) it can be $O(n)$.
- **Longest Common Subseq.:** Eg. ("abcd", "bcd") \Rightarrow "bcd". Structure $\mathcal{P}(x_{0:i}, y_{0:j})$. Let $A_{i,j}$ be LCS between $x_{0:i}, y_{0:j}$:

$$A_{i,j} = \begin{cases} A_{i-1,j-1} + 1 & x_i = y_j \\ \max\{A_{i,j-1}, A_{i-1,j}\} & x_i \neq y_j \end{cases}$$

Can pad a zero row & column as $i, j = -1$. Proceed normally.

- **Longest Increasing Subseq.:** Eg. [2,5,3,1,4] \Rightarrow [2,3,4]. Has substructure $\mathcal{P}(x_{0:i})$ while single state transition is $O(n)$, which gives $O(n^2)$ in total. Let A_i be the LIS between $[0, i]$ and ends with x_i . Target is $\max_i \{A_i\}$. $A_i = \max_{0 \leq j < i} \{\mathbb{1}_{\{x_j < x_i\}} A_j + 1\}$, initial condition $A_0 = 1$. This problem turns out to have $O(n \log n)$ *alternative solution*.
- **Longest Palindromic Subseq.:** Eg. "adbcbca" \Rightarrow "abcba". This follows structure $\mathcal{P}(x_{i:j})$. Solved by $O(n^2)$ DP with 2d-table \mathbf{A} : $A_{i,j} :=$ Length of LPS within $[i, j]$ of s . Target is $A_{0,n}$.

$$A_{i,j} = \begin{cases} A_{i+1,j-1} + 2 & s_i = s_j \\ \max\{A_{i,j-1}, A_{i+1,j}\} & s_i \neq s_j \end{cases}$$

Boundary condition: $A_{i,i} = 1$ on main diagonal, proceed \nearrow .

- **Knapsack Problems:**
 - **0-1 Knapsack:** $1 \leq k \leq K$ items with weights W_k and value V_k . Put items in a knapsack of capacity N to achieve maximum total value. Let $f(k, n)$ be the solution of using first k items to fill capacity n . Then:

$$f(k, n) = \begin{cases} 0; & \text{If } k = 0 \text{ or } n = 0 \\ \max\{f(k-1, n), f(k-1, n - W_k) + V_k\} & \end{cases}$$

For the k -th item, there's really 2 states: (1) Not keeping it, the max value is same as $f(k-1, n)$. (2) Keep it, last state with first $(k-1)$ items has at least $n - W_k$ capacity, resulting in value $f(k-1, n - W_k) + V_k$. Time complexity: $O(KN)$.

Disguised Application: **LC-474**.

- **Complete Knapsack:** Ask how many different *ways* to *completely* fill bag capacity with any number of (copies of) items. Let $f(k, n)$ be the number of ways only using first k ($k \geq 1$) items to fill capacity n .

$$f(k, n) = \begin{cases} 1 & n = 0 \\ f(k-1, n) + f(k, n - W_k) & n - W_k \geq 0 \\ f(k-1, n) & n - W_k < 0 \end{cases}$$

Think recursively: there are two set of "ways" to reach state (k, n) : (1) use first $k-1$ items to fill n , $f(k-1, n)$. (2) use 1 extra unit of k -th item, $f(k, n - W_k)$; only doable when $n - W_k \geq 0$.

Disguised Application: **LC-518: Coin Change II**.

- **Subset 0-1 Knapsack:** Determine whether there's a subset of items, whose weight sum = capacity N .

$$\text{Let } \mathbb{1}_{k,n} = \mathbb{1} \{ \exists S \subset \text{items}[0:k] : \sum_{j \in S} W_j = n \}$$

$$\mathbb{1}_{k,n} = \begin{cases} 1 & n = 0 \\ 0 & n > 0, k = 0 \\ \mathbb{1}_{k-1,n} & n - W_k < 0 \\ \max\{\mathbb{1}_{k-1,n}, \mathbb{1}_{k-1,n-W_k}\} & n - W_k \geq 0 \end{cases}$$

Application: **LC-416: Partition Equal Subset Sum**.

- **K Eggs:** Follow up on the 2-Egg problem. Let $f(n, k)$ be the optimal throws for the subproblem (n, k) . Drop the first egg on floor m . If it breaks, the remaining problem is $f(m-1, k-1)$, otherwise $f(n-m, k)$. DP recursion is: $f(n, k) = \min_{1 \leq m < n} \{1 + \max(f(m-1, k-1), f(n-m, k))\}$

- **Dynamic Card Game:** (*Green Book pp.127*) There are 50 red and 50 black cards, draw at most 100 times. Early stop gives payoff $= r/(b+r)$, i.e. ratio of red cards at hand. Find stopping policy and expection of the game. Let the expected payoff be $f(r, b)$ standing at state (r, b) , perform DP:

$$f(r, b) = \max \left\{ \frac{r}{r+b}, \frac{n-r}{n-r-b} f(r+1, b) + \frac{n-b}{n-r-b} f(r, b+1) \right\}$$

The boundary conditions are $f(r, 50) = 1/2$ (just draw till the 50,50 state when there's no black card in the deck) and $f(50, b) = \frac{50}{b+50}$ (stop when there's no red card in the desk). Proceed \searrow in the DP table and return $f(0, 0)$.

- **Buy and Sell Stocks IV:** (The general case) Given price series $p_{1:n}$, trade m times, hold 1 position at a time. The hard part is defining the states of the DP, which turns out to be $(t, k, \mathbb{1}_{\text{pos}})$: at time t , have already traded k time in $1:k-1$, and whether holding a position or not (0/1). Let $f(t, k, \mathbb{1}_{\text{pos}})$ be the max PnL of this state, DP transition is:

$$f(t, k, 1) = \max \left\{ \underbrace{f(t-1, k-1, 0) - p_t}_{\text{Buy at } t}, \underbrace{f(t-1, k, 1)}_{\text{Rest}} \right\}$$

$$f(t, k, 0) = \max \left\{ \underbrace{f(t-1, k, 1) + p_t}_{\text{Sell at } t}, \underbrace{f(t-1, k, 0)}_{\text{Rest}} \right\}$$

- (Optimization for large m) If $m > n/2$, reduces to the special case with unlimited number of trade. Should be checked before launching huge DP array for large m : Summing up all upticks.
- (Special case) **Buy & Sell Stocks I** trade once. Let $f(k)$ be the max PnL over $1:k$, keep track of historical min till k yields DP $f(k) = \max\{f(k-1), p_k - p_{\min, 1:k}\}$ $O(n)$ time and $O(1)$ space. This is the *mirror problem* of **Max Drawdown**.

- **House Robber:**
- **World Series:** (*Green Book, pp.123*)

10.5 Backtrack

10.6 Misc

- **Prime number Sieve:** Time complexity: $O(n \log \log n)$. Optimizations: for number p , sieve out $p^2, p^2 + p, \dots$, instead of $2p, 3p, \dots$. And only need to screen $p \leq \sqrt{n}$.
- **Exponentiation:** Note the recursion: $a^b = (a^{b/2})^2$ for b even. $a^{b/2}$ need to be computed only once.
- **Settings:**

```
1: function RATINGTRANS( $\mathbf{P}, T, \mathcal{M}_\tau, n, d$ ) //  $\mathcal{M}_\tau$  is  $\tau$ 's sampler
2:    $\mathbf{\Pi} \leftarrow \text{zeros}((d, d))$  //  $\text{table}[i]$  is sampler for  $X(\tau + dt) | X(\tau) = i$ 
3:    $\text{tables} \leftarrow [\text{MakeTable}(\text{zip}(\text{range}(d), \mathbf{P}[i])) \text{ for } i \text{ in } \text{range}(d)]$ 
4:   for  $x_0$  in  $\text{range}(d)$  do // simulate  $x_0$ -th row of  $\mathbf{\Pi}$ 
5:     for  $i = 1 : n$  do // draw  $n$  samples
6:        $t, x \leftarrow 0, x_0$ 
7:       while  $t \leq T$  do
8:          $\tau \leftarrow \mathcal{M}_\tau(x); t += \tau$ 
9:         if  $t > T$  then break while // transition exceeds  $T$ 
10:         $x \leftarrow \text{Draw}(\text{tables}[x])$  // sample next state
11:         $\mathbf{\Pi}[x_0, x] += 1$  // while ends,  $x$  is the final state
12:    $\mathbf{\Pi} ./= n; se \leftarrow \sqrt{\mathbf{P}\mathbf{i}(1 - \mathbf{P}\mathbf{i})/n}$ 
13:   return  $\mathbf{\Pi}, se$ 
```

11 C++ (Summary Repo)

- **OOP Property:** Encapsulation, Inheritance, Polymorphism.
- **C++ Polymorphism:** Compile-time: function/operator overloading, function/class templates. Same name \rightarrow multiple signatures & implementations. Run-time: subtype polymorphism; The ability to use derived classes through base class ptrs and refs.
- **Virtual Function:** makes the base class member function overridden by that of derived class. In particular, find the derived class function via pointer-to-base-class (that actually points to a derived instance). `pbase->f()` calls `derived.foo`.
- **Virtual Base Class:** Consider a diamond-like multiple inheritance $A \rightarrow B, C \rightarrow D$. Duplicate base class A will be created in memory layout. Letting B, C inherit virtually from A will assure single A .
- **New/Delete/Malloc/Free:** `malloc/free` are `stdlib` function of C and Cpp. `new/delete` are Cpp operators. The first pair only allocate/deallocate memory, while the latter pair also invokes constructor and destructor of dynamically allocated objects.
- **The Rule of Three:** When implement a class that manage a resource: implement Copy, copy-assignment, destructor. Rule-of-Five after C++ 11: Move, move-assignment.
 - Default copy and copy-assignment: memberwise copy of its sub-objects.
 - Default destructor: after executing the body of destructor and destroying automatic objects allocated within the body, destructor of X calls the destructor of X 's direct members.
- **Move Semantics:** Copy the pointer from source, and set the original pointer in the source to null - Steal the data from source (without creating copy). Here, source is an *rvalue*, the client won't inspect its content ever again.

12 Python

- **Pass By Object Reference:** Behave like "pass by a copy of pointer to the object". It's a reference (name binded) to the object, but a separate one. Suppose $x = [1, 2, 3]$, inside function scope:
 - `x.append(4)` It's changing the object referenced by name x . Change will be observed outside the function.
 - `x = [4, 5, 6]` It's re-binding the name x "a copied pointer" to a new object. No change outside the function.
- If x binds to an immutable object, nothing can be changed inside the function.
- **Garbage Collection:** Reference count mechanism, similar to the `smart_pointer` in Cpp. CPython: There is a `PyObject` struct underneath python objects, that maintains reference count. Caveat: *Circular Referencing*, GC will make a directed graph of the cross-referencing relationship. Check parents whenever ref count = 0 for some node.
- **Reflection:** Being able to examine and modify its own structure in Run-Time: `getattr()`, `setattr()`, `type()`, `dir()`.
- **Context Manager:** `__enter__`, `__exit__`.
- **Copy Object:** `copy.copy(...)`; use `copy.deepcopy(...)` to copy the encapsulated object within the compound object.

- **Decorator:** Syntax sugar, a function that takes the decorated function as one arg, and returns a transformed function.
- **Metaclass:** A derived class of `type`, whose instance can be used as a class.

13 Database

- **KDB:**
 - **Representation:** Columnar representation of data. Columns are contiguous on memory, allowing vectorized operation & treating columns as atomic data units.
 - **Storage:** Memory-mapped file on disk. No need to translate data from disk to RAM, faster IO.
 - **Language:** Efficient query language, small footprint, resides on L1/2 cache of CPU.
- **SQL:**
 - `JOIN ON = pd.merge`: inner join by default. Left, right, outer join. Make sure non-duplicate values in the join-key column.
 - `UNION = pd.concat`.
 - `GROUP BY = df.groupby()`. `HAVING`: filter the output of group by aggregation.
 - **Second Largest from Groups:** `ROW_NUMBER() OVER (PARTITION BY <groupId> ORDER BY <val> DESC)`.
 - **Find Duplicate:** `SELECT col, COUNT(*) FROM t GROUP BY col HAVING COUNT(*) > 1`. To extract the duplicate rows, join this agg table to the original table.
 - `CASE WHEN`: transform numerical to categorical.

14 Macroeconomics

•

15 Resume (2020 Spring)

- **Russell Rebalance:** Obtained from CS research desk the point-in-time rule-based estimates of the projected indexer's flow at the reconstitution date, what we call the "trade file".
 - **Goal:** Build a point-in-time estimate of the prepositioners' cumulative inventory trajectory, with which we can compare against the trade file to decide: (1) whether it's overplayed/underplayed, hence the backside direction; (2) timing to start to enter rebal positions; (3) arbitrage the arbitrageurs.
 - **Portfolio Execution Detection Problem:** The prepositioners trade a fraction of the indexer flow with an opposite sign, which creates a cross-sectional signal.
 - **Relevant Features:** *Intraday Raw Message and LOB Snaps*, mainly the 5-min bar aggregation, leveraged cloud computing environment. *Short interest* is useful for the short-only tranche, say R2 deletion. In 2014, for example, it was announced that all Business Development Companies will be removed from Russell. Observe a big spike of SI/SharesOutstanding as early as in March for BDCs.
 - **Closing Auction:** Also plays an important part: (1) cross-sectional signal of prepositioners also exists. (2) Metric/estimate of indexers & prepositioners trade on the reconstitution date. Use this to train/validate the model.
 - **Result Numbers:**
- **Market Impact Model:**
 - **Goal:**
 - **Propagator Model:** Exogenous unaffected price $P^0(t)$ be an arithmetic Brownian motion whatsoever. Affected price:

$$P^x(t) = P^0(t) + \int_{-\infty}^t f(\dot{x}(s), z_s) G(t-s) ds + h(\dot{x}(t), z_t) \quad (1)$$

Sum of 3 things: unaffected price, transient impact (spread, financing cost, etc), and convolution of the instantaneous impact of all child orders before t with a decay kernel G .

- **Decay Kernel:** In particular $G = 1$, the convolution will be linear in t , which means $f(\dot{x}(s), z_s)$ stacks on price permanently. This is "permanent impact" in the Almgren-Chriss model.
- **Volume Model:** AR2 Model. Asymmetric return effect: negative return correlates increase in vol hence increase in volume (leverage effect). Special dates adjustment.
- **U-Curve Model:** Let standardized cumulative u-curve $\tau(t) = \int_0^t v(s) ds / V_d$, and daily factor $x_{d,i}$. This is a functional regression problem with function response:

$$\tau_{d,i}(t) = \beta_0(t) + \beta_1(t)x_{d,i} + u_{d,i}(t)$$

For 5-min bars, $\beta_1(t)$: linear interpolation among 5-min spaced knots, 78 parameters. Dimension reduction:

$$\tau_{d,i}(t) = \bar{\tau}(t) + s_{d,i,1}v_1(t) + \dots + s_{d,i,k}v_k(t) + u_{d,i}(t)$$

where $v_1(t), \dots, v_k(t)$ are PC directions. It then suffices to estimate $s_{d,i,k} = \beta_k x_{d,i} + \epsilon_{d,i,k}$, i.e. the relationship between factors and PC scores.

- **U-Curve PCA:** Leading 3 PCs are stable over time, 75% variance explained. Note: PCs for cumulative u-curve. Imagine the derivative of them to be actual effect on u-curve. PC1: "Front-loadedness" (+-); PC2: "Steepness" (+ - +); PC3: (- + -+).

- **Cross-Impact:** write (1) in its multivariate version:

$$\int_{-\infty}^t \text{diag}\{G(t-s)\}(\lambda \mathbf{\Gamma} + (1-\lambda)\mathbf{I})\mathbf{f}(\mathbf{x}(s), \mathbf{z}_s)ds.$$
Where $\mathbf{\Gamma}$ can be some analogous form of intraday correlation matrix. *If asked: We end up with something induced by an intraday PCA factor model plus a block diagonal that represents GICs industries. Variance > Bias most of the time, for a lot of strategies univariate model is sufficient. But do want cross-impact for strategies that trade correlated names in their nature: pairs trading.*

- **Result Numbers:** R^2 : Sub 1% \rightarrow 2.5% against slippage to arrival of metaorders. The model's been applied in the largest systematic book which runs mid-frequency mean-reversion strategy. Dollar weighted slippage reduced by 0.6BPs (3.7 \rightarrow 3.1BPs), equivalent to about 1.5M dollar revenue per year for 500×500 M L/S book 10% daily turnover.

- **Infrastructure:** This is more of a research-infrastructure-project so to speak. *In fact it's not even a standalone project but a side-product developed from other stuffs I've done. And turns out to be very good and well-received by other people.*

- **Goal:** It tries to answer the methodological or philosophical question, that how data sources, features, and intermediate results should be organized for the models or statistical procedures that may have *many different versions*.

Feed Forward Computational Graph: Imagine all (including intermediate) variables used in simulation constitute a directed graph. If we topologically sort these variables and evaluate sequentially, we can evaluate the model partially and avoid repeating same computations. So we don't have to keep track of different versions of models/simulations, but just collect everything (the full model) together in the graph and evaluate partially.

- **Earning Momentum:**

- **Event Study Regression:** Cumulative abnormal return subsequent to the announcement date (POSTCAR) \sim Surprise metrics + control factors.
- **Surprise Metric - Consensus Error:** Standard Unexpected Earning, more generally, *Standardized Consensus Error*:

$$\text{SCE} = \frac{\text{Mean Est.} - \text{Actual}}{20\text{-days Avg.Price}}$$

The underlying assumption is: consensus forecast is an unbiased measure of the market's expected earnings. But is biased: *Agency problem*, many analysts are engaged in earning-guidance game such that they overshoot at the start of the year, then lower down to a level that the firm can actually beat.

- **Surprise Metric - Difference in Fraction of Misses:**

$$\text{DFM} = \frac{\#\text{Est} > \text{A}}{N} - \frac{\#\text{Est} < \text{A}}{N} = \text{Mean}(\text{sgn}(E_i - A_i))$$

Is immune to biases and outliers. Beats SUE in R^2 by 30%.

- **Revision Metric:** 6-month moving average of changes in consensus forecast:

$$\text{REV6} = \frac{1}{6} \sum_{j=0}^5 \frac{\text{Mean}(E_{t-j}) - \text{Mean}(E_{t-j-1})}{P_{t-j-1}}$$

Mostly negative: another evidence that they overshoot at the beginning.

- **3-Way Sort:** Earning, revenue, and price momentum tend to affect the CAR jointly - eg. revenue momentum is significant on its own, but fades after adding earning and price - in fact all 3 effects decreased in joint regression, but still have independent contribution.
- **Size:** Momentum profits decline with size: small firm slow information diffusion, hence greater momentum and slower decay.
- **Analyst Coverage:** mainly explained by size; as well as book/market, turnover, index membership. But size dominates.

$$\text{COV} \sim \log(\text{SIZE}) + B/M + \text{TVR} + \mathbb{1}_{R_i} + u_{\text{COV}}$$

- **Residual Analyst Coverage:** Control the size - then momentum premium is larger for low *residual analyst coverage*, i.e. the residual of the COV model above regressed only on SIZE.
- **Result Numbers:** Realized Sharpe = 1.3 in 2018Q3 - 2019Q4. This year since the COVID-19 outbreak, most systematic strategies are truncate to minimal size or 0.

16 Questions

- **(Misc):** Momentum. Carhart's 4-factor model: past 12 months average return. Barra: Relative strength descriptor, weighted sum of excess (minus risk free) log returns over the trailing 504 trading days (2 years). w_s are exponential weights with half-life = 126 days.

$$\text{RSTR}(t) = \sum_{s=1}^{504} w_s (\ln(1 + r_{t-s}) - \ln(1 + f_{t-s}))$$

- **(Misc):** NY population: 8M. US population: 330M.
- Briefly describe a typical project of the group. How does that fit into the entire systematic trading picture? How does the project make impact and receive feedbacks.
- Name a book in mathematics and statistics that you find to be most useful for the role / or you continue to refer to it every once a while.
-

$$\begin{aligned} & \text{Min}_{\mathbf{x}} \frac{1}{2} \mathbf{x}^\top \mathbf{D} \mathbf{x}, \quad \text{s.t. } \mathbf{B} \mathbf{x} = \mathbf{e} \\ \Rightarrow & \mathbf{D} \mathbf{x} = \mathbf{B}^\top \mathbf{u}^\top; \quad \mathbf{B} \mathbf{x} = \mathbf{e} \\ \Rightarrow & \mathbf{u}^\top = (\mathbf{B} \mathbf{D}^{-1} \mathbf{B}^\top)^{-1} \mathbf{e} \\ \Rightarrow & \mathbf{x} = \mathbf{D}^{-1} \mathbf{B}^\top (\mathbf{B} \mathbf{D}^{-1} \mathbf{B}^\top)^{-1} \mathbf{e} \end{aligned} \tag{2}$$