

1. 决策树 (Decision Tree)
 2. 期望值 (Expected Value)
 3. 风险 (Risk)
 4. 效用 (Utility)
 5. 决策规则 (Decision Rule)

1. 期望值 (Expected Value)

$$E(X) = \sum_{i=1}^n p_i x_i$$
 2. 风险 (Risk)

$$V(X) = \sum_{i=1}^n p_i (x_i - E(X))^2$$
 3. 效用 (Utility)

$$U(X) = \sum_{i=1}^n p_i u(x_i)$$
 4. 决策规则 (Decision Rule)

1. 决策树 (Decision Tree)
 2. 期望值 (Expected Value)
 3. 风险 (Risk)
 4. 效用 (Utility)
 5. 决策规则 (Decision Rule)

1. 决策树 (Decision Tree)
 2. 期望值 (Expected Value)
 3. 风险 (Risk)
 4. 效用 (Utility)
 5. 决策规则 (Decision Rule)



Handwritten:

D	P	N
P	a	b
N	c	d

aTP
bTP
cTP
dTP

(D) (P) (N)
A C G U
U A C G

① 有錯 (ML 8/2/22)

114/41

21/11

$f_2(x) = \frac{1}{2}x^2$

$\begin{matrix} P & (P) a & (P) b \\ W & (P) c & (P) d \end{matrix}$

$P = \frac{a+b+c+d}{n}$
 $R = \frac{a+b}{n}$
 $C = \frac{a+c}{n}$
 $D = \frac{b+d}{n}$

$Accuracy = \frac{a+d}{n}$
 $Precision = \frac{a}{a+c}$
 $Recall = \frac{a}{a+b}$
 $F_1 = \frac{2PR}{P+R}$

$f(z) = \frac{1}{1+e^{-z}}$
 $z = \beta_0 + \beta_1 x_1 + \dots + \beta_n x_n$
 $P(y=1|z) = f(z)$
 $P(y=0|z) = 1 - f(z)$

$\frac{TP}{TP+FN}$
 $\frac{FP}{FP+FN}$
 $F_1 = \frac{2PR}{P+R}$



Data:
 11111 + 1234
 123456789101112
 13141516171819202122232425262728293031323334353637383940414243444546474849505152535455565758596061626364656667686970717273747576777879808182838485868788899091929394959697989910010110210310410510610710810911011111211311411511611711811912012112212312412512612712812913013113213313413513613713813914014114214314414514614714814915015115215315415515615715815916016116216316416516616716816917017117217317417517617717817918018118218318418518618718818919019119219319419519619719819920020120220320420520620720820921021121221321421521621721821922022122222322422522622722822923023123223323423523623723823924024124224324424524624724824925025125225325425525625725825926026126226326426526626726826927027127227327427527627727827928028128228328428528628728828929029129229329429529629729829930030130230330430530630730830931031131231331431531631731831932032132232332432532632732832933033133233333433533633733833934034134234334434534634734834935035135235335435535635735835936036136236336436536636736836937037137237337437537637737837938038138238338438538638738838939039139239339439539639739839940040140240340440540640740840941041141241341441541641741841942042142242342442542642742842943043143243343443543643743843944044144244344444544644744844945045145245345445545645745845946046146246346446546646746846947047147247347447547647747847948048148248348448548648748848949049149249349449549649749849950050150250350450550650750850951051151251351451551651751851952052152252352452552652752852953053153253353453553653753853954054154254354454554654754854955055155255355455555655755855956056156256356456556656756856957057157257357457557657757857958058158258358458558658758858959059159259359459559659759859960060160260360460560660760860961061161261361461561661761861962062162262362462562662762862963063163263363463563663763863964064164264364464564664764864965065165265365465565665765865966066166266366466566666766866967067167267367467567667767867968068168268368468568668768868969069169269369469569669769869970070170270370470570670770870971071171271371471571671771871972072172272372472572672772872973073173273373473573673773873974074174274374474574674774874975075175275375475575675775875976076176276376476576676776876977077177277377477577677777877978078178278378478578678778878979079179279379479579679779879980080180280380480580680780880981081181281381481581681781881982082182282382482582682782882983083183283383483583683783883984084184284384484584684784884985085185285385485585685785885986086186286386486586686786886987087187287387487587687787887988088188288388488588688788888989089189289389489589689789889990090190290390490590690790890991091191291391491591691791891992092192292392492592692792892993093193293393493593693793893994094194294394494594694794894995095195295395495595695795895996096196296396496596696796896997097197297397497597697797897998098198298398498598698798898999099199299399499599699799899910001001100210031004100510061007100810091010101110121013101410151016101710181019102010211022102310241025102610271028102910301031103210331034103510361037103810391040104110421043104410451046104710481049105010511052105310541055105610571058105910601061106210631064106510661067106810691070107110721073107410751076107710781079108010811082108310841085108610871088108910901091109210931094109510961097109810991100110111021103110411051106110711081109111011111112111311141115111611171118111911201121112211231124112511261127112811291130113111321133113411351136113711381139114011411142114311441145114611471148114911501151115211531154115511561157115811591160116111621163116411651166116711681169117011711172117311741175117611771178117911801181118211831184118511861187118811891190119111921193119411951196119711981199120012011202120312041205120612071208120912101211121212131214121512161217121812191220122112221223122412251226122712281229123012311232123312341235123612371238123912401241124212431244124512461247124812491250125112521253125412551256125712581259126012611262126312641265126612671268126912701271127212731274127512761277127812791280128112821283128412851286128712881289129012911292129

[illegible]

② 排序
 按址: ①: 逐点的比较
 ②: 逐点比较
 ③: 逐点比较
 ④: 逐点比较
 ⑤: 逐点比较
 ⑥: 逐点比较
 ⑦: 逐点比较
 ⑧: 逐点比较
 ⑨: 逐点比较
 ⑩: 逐点比较
 ⑪: 逐点比较
 ⑫: 逐点比较
 ⑬: 逐点比较
 ⑭: 逐点比较
 ⑮: 逐点比较
 ⑯: 逐点比较
 ⑰: 逐点比较
 ⑱: 逐点比较
 ⑲: 逐点比较
 ⑳: 逐点比较
 ㉑: 逐点比较
 ㉒: 逐点比较
 ㉓: 逐点比较
 ㉔: 逐点比较
 ㉕: 逐点比较
 ㉖: 逐点比较
 ㉗: 逐点比较
 ㉘: 逐点比较
 ㉙: 逐点比较
 ㉚: 逐点比较
 ㉛: 逐点比较
 ㉜: 逐点比较
 ㉝: 逐点比较
 ㉞: 逐点比较
 ㉟: 逐点比较
 ㊱: 逐点比较
 ㊲: 逐点比较
 ㊳: 逐点比较
 ㊴: 逐点比较
 ㊵: 逐点比较
 ㊶: 逐点比较
 ㊷: 逐点比较
 ㊸: 逐点比较
 ㊹: 逐点比较
 ㊺: 逐点比较
 ㊻: 逐点比较
 ㊼: 逐点比较
 ㊽: 逐点比较
 ㊾: 逐点比较
 ㊿: 逐点比较

④ 克拉夫特不等式 (Kraft inequality)

设 $\{p_i\}$ 为概率分布，则

① 当且仅当 $\sum_{i=1}^{\infty} p_i^2 \leq 1$ 时，存在唯一的前缀码。

② 当且仅当 $\sum_{i=1}^{\infty} p_i^2 = 1$ 时，存在唯一的前缀码。

③ 当且仅当 $\sum_{i=1}^{\infty} p_i^2 > 1$ 时，不存在前缀码。

$$= \sigma_{y|x}^2 \sum_{i=1}^n \left(t_i + \frac{1}{n} \sum_{j=1}^n x_j \right)^2 - \frac{1}{n} \sum_{j=1}^n x_j^2$$

$$e'(t) = e(t) + \lambda x_0 \cdot \frac{d}{dt} e(t) \quad \text{N: 对于连续时间}$$

$$e(t) = \frac{1}{1 + \lambda x_0} \cdot e(t) \quad \text{N: 对于离散时间}$$

③ Bayes: $P(C|A_1 \dots A_n) = \frac{P(A_1 \dots A_n | C) P(C)}{P(A_1 \dots A_n)}$

④ 卡方检验: 多组数据: 卡方检验中显著性 = 0.05

$\Delta \rho_{\text{pr}} = \frac{\Delta \rho_{\text{pr}}}{\rho_{\text{pr}}} = \frac{\Delta \rho_{\text{pr}}}{\rho_{\text{pr}}}$

① 线性回归
 ② 非线性回归
 ③ 逻辑回归
 ④ 支持向量机
 ⑤ 决策树
 ⑥ 随机森林
 ⑦ 神经网络
 ⑧ 深度学习
 ⑨ 迁移学习
 ⑩ 强化学习

③ SVM 通过寻找超平面 (hyperplane) 来区分数据

max γ
s.t. $y_i (\frac{w}{\|w\|} \cdot x_i + \frac{b}{\|w\|}) \geq \gamma \Rightarrow \begin{cases} \min \frac{1}{2} \\ y_i (w \cdot x_i + b) \geq \gamma \end{cases}$

Large margin is the:

$L(w, b, \alpha) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^n \alpha_i (y_i (w \cdot x_i + b) - 1)$

求解 $\min_{w, b, \alpha} L(w, b, \alpha) = p \Rightarrow \begin{cases} w = \sum_{i=1}^n \alpha_i y_i x_i \\ \sum_{i=1}^n \alpha_i = 1 \end{cases}$

$$d(C_i) = \frac{1}{P} \left(\sum_{j \in \text{ord}(C_i)} d(C_j) + d(C_{i+1}) \right)$$

$(\frac{K(pn)}{2})^n$ 个, 不连续
 $|x_{ip} - x_{jp}| = |x_{ij}|$
 $\frac{1}{2} \sum_{i,j} |x_{ij}|$
 $j = \frac{b^2 c}{a+b+c}$
 $\frac{1}{2} \sum_{i,j} |x_{ij}|$
 $\frac{1}{2} \sum_{i,j} |x_{ij}|$
 $\frac{1}{2} \sum_{i,j} |x_{ij}|$

$f(x) = \text{sign}(w \cdot x + b) = \text{sign}(\sum_i w_i x_i + b)$
 $\text{sign}(w \cdot x + b) = \begin{cases} 1 & \text{if } w \cdot x + b \geq 0 \\ -1 & \text{if } w \cdot x + b < 0 \end{cases}$
 $\text{sign}(w \cdot x + b) = \begin{cases} 1 & \text{if } \sum_i w_i x_i + b \geq 0 \\ -1 & \text{if } \sum_i w_i x_i + b < 0 \end{cases}$
 $\text{sign}(w \cdot x + b) = \begin{cases} 1 & \text{if } \sum_i w_i x_i + b \geq 0 \\ -1 & \text{if } \sum_i w_i x_i + b < 0 \end{cases}$

K-means:

$$E = \sum_{i=1}^n \sum_{j=1}^K |p_i - \mu_j|^2$$

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. 26. 27. 28. 29. 30. 31. 32. 33. 34. 35. 36. 37. 38. 39. 40. 41. 42. 43. 44. 45. 46. 47. 48. 49. 50. 51. 52. 53. 54. 55. 56. 57. 58. 59. 60. 61. 62. 63. 64. 65. 66. 67. 68. 69. 70. 71. 72. 73. 74. 75. 76. 77. 78. 79. 80. 81. 82. 83. 84. 85. 86. 87. 88. 89. 90. 91. 92. 93. 94. 95. 96. 97. 98. 99. 100. 101. 102. 103. 104. 105. 106. 107. 108. 109. 110. 111. 112. 113. 114. 115. 116. 117. 118. 119. 120. 121. 122. 123. 124. 125. 126. 127. 128. 129. 130. 131. 132. 133. 134. 135. 136. 137. 138. 139. 140. 141. 142. 143. 144. 145. 146. 147. 148. 149. 150. 151. 152. 153. 154. 155. 156. 157. 158. 159. 160. 161. 162. 163. 164. 165. 166. 167. 168. 169. 170. 171. 172. 173. 174. 175. 176. 177. 178. 179. 180. 181. 182. 183. 184. 185. 186. 187. 188. 189. 190. 191. 192. 193. 194. 195. 196. 197. 198. 199. 200. 201. 202. 203. 204. 205. 206. 207. 208. 209. 210. 211. 212. 213. 214. 215. 216. 217. 218. 219. 220. 221. 222. 223. 224. 225. 226. 227. 228. 229. 230. 231. 232. 233. 234. 235. 236. 237. 238. 239. 240. 241. 242. 243. 244. 245. 246. 247. 248. 249. 250. 251. 252. 253. 254. 255. 256. 257. 258. 259. 260. 261. 262. 263. 264. 265. 266. 267. 268. 269. 270. 271. 272. 273. 274. 275. 276. 277. 278. 279. 280. 281. 282. 283. 284. 285. 286. 287. 288. 289. 290. 291. 292. 293. 294. 295. 296. 297. 298. 299. 300. 301. 302. 303. 304. 305. 306. 307. 308. 309. 310. 311. 312. 313. 314. 315. 316. 317. 318. 319. 320. 321. 322. 323. 324. 325. 326. 327. 328. 329. 330. 331. 332. 333. 334. 335. 336. 337. 338. 339. 340. 341. 342. 343. 344. 345. 346. 347. 348. 349. 350. 351. 352. 353. 354. 355. 356. 357. 358. 359. 360. 361. 362. 363. 364. 365. 366. 367. 368. 369. 370. 371. 372. 373. 374. 375. 376. 377. 378. 379. 380. 381. 382. 383. 384. 385. 386. 387. 388. 389. 390. 391. 392. 393. 394. 395. 396. 397. 398. 399. 400. 401. 402. 403. 404. 405. 406. 407. 408. 409. 410. 411. 412. 413. 414. 415. 416. 417. 418. 419. 420. 421. 422. 423. 424. 425. 426. 427. 428. 429. 430. 431. 432. 433. 434. 435. 436. 437. 438. 439. 440. 441. 442. 443. 444. 445. 446. 447. 448. 449. 450. 451. 452. 453. 454. 455. 456. 457. 458. 459. 460. 461. 462. 463. 464. 465. 466. 467. 468. 469. 470. 471. 472. 473. 474. 475. 476. 477. 478. 479. 480. 481. 482. 483. 484. 485. 486. 487. 488. 489. 490. 491. 492. 493. 494. 495. 496. 497. 498. 499. 500. 501. 502. 503. 504. 505. 506. 507. 508. 509. 510. 511. 512. 513. 514. 515. 516. 517. 518. 519. 520. 521. 522. 523. 524. 525. 526. 527. 528. 529. 530. 531. 532. 533. 534. 535. 536. 537. 538. 539. 540. 541. 542. 543. 544. 545. 546. 547. 548. 549. 550. 551. 552. 553. 554. 555. 556. 557. 558. 559. 560. 561. 562. 563. 564. 565. 566. 567. 568. 569. 570. 571. 572. 573. 574. 575. 576. 577. 578. 579. 580. 581. 582. 583. 584. 585. 586. 587. 588. 589. 590. 591. 592. 593. 594. 595. 596. 597. 598. 599. 600. 601. 602. 603. 604. 605. 606. 607. 608. 609. 610. 611. 612. 613. 614. 615. 616. 617. 618. 619. 620. 621. 622. 623. 624. 625. 626. 627. 628. 629. 630. 631. 632. 633. 634. 635. 636. 637. 638. 639. 640. 641. 642. 643. 644. 645. 646. 647. 648. 649. 650. 651. 652. 653. 654. 655. 656. 657. 658. 659. 660. 661. 662. 663. 664. 665. 666. 667. 668. 669. 670. 671. 672. 673. 674. 675. 676. 677. 678. 679. 680. 681. 682. 683. 684. 685. 686. 687. 688. 689. 690. 691. 692. 693. 694. 695. 696. 697. 698. 699. 700. 701. 702. 703. 704. 705. 706. 707. 708. 709. 710. 711. 712. 713. 714. 715. 716. 717. 718. 719. 720. 721. 722. 723. 724. 725. 726. 727. 728. 729. 730. 731. 732. 733. 734. 735. 736. 737. 738. 739. 740. 741. 742. 743. 744. 745. 746. 747. 748. 749. 750. 751. 752. 753. 754. 755. 756. 757. 758. 759. 760. 761. 762. 763. 764. 765. 766. 767. 768. 769. 770. 771. 772. 773. 774. 775. 776. 777. 778. 779. 780. 781. 782. 783. 784. 785. 786. 787. 788. 789. 790. 791. 792. 793. 794. 795. 796. 797. 798. 799. 800. 801. 802. 803. 804. 805. 806. 807. 808. 809. 810. 811. 812. 813. 814. 815. 816. 817. 818. 819. 820. 821. 822. 823. 824. 825. 826. 827. 828. 829. 830. 831. 832

$\lim_{n \rightarrow \infty} \frac{1}{n} \sum_{j=1}^n d(x_j, y_j)^2 = \frac{1}{n} \sum_{j=1}^n d(x_j, y_j)^2$
 收敛中心
 每次找出的各个点
 中心 = 各个点的平均
 直到不再为止
 且中心收敛到所有
 外点的平均

[illegible]

$\frac{1}{\sqrt{2}}$

news: 一个 cluster
每个性能又差又慢, 没有冗余的备份
ad: bagging bootstrap
bagging 有同义词 \Rightarrow bootstrap 若干, 多个小
(把数据打乱) 数据, 多次合为一起 (多个数据)
bootstrap: 打乱数据池 (随机生成)
effect: 影响

$$BSS = \sum_{i=1}^n |c_i| (a_i - b_i)^2 \quad \text{Total} = \sum_{i=1}^n \frac{10 \cdot 1}{15} \times \text{priority}(b_i)$$

max: 波長 (nm)
(1) 波長: 400nm
波長: 400nm, 合共 400nm

大 学 生 的 打 工 经 历
 Student's Work Experience
 姓 名 (姓 名 姓 名)
 Name (Name Name)

KB: knowledge base

modus ponens $\frac{A \quad A \rightarrow B}{B}$

or-introduction $\frac{A}{A \vee B}$

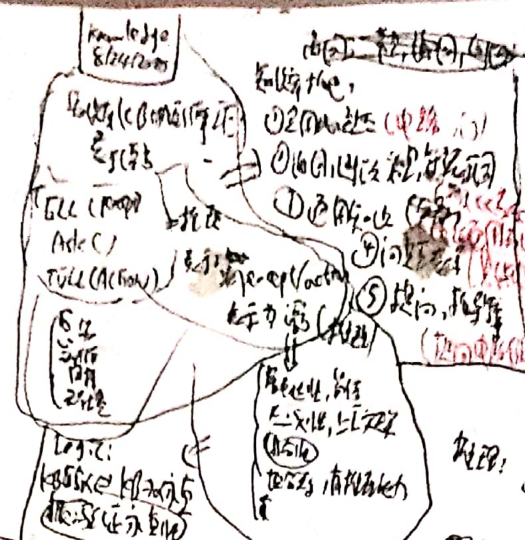
and-elimination $\frac{A \wedge B}{A}$

double negation elimination $\frac{\neg \neg A}{A}$

and-introduction $\frac{A \quad B}{A \wedge B}$

resolution $\frac{A \vee B \quad \neg A \vee C}{B \vee C}$

推理: (前提) (结论)



Goal: G

Initial state: S_0

Successor function: $SUC(S, a)$

Path cost: $c(a)$

Goal test: $G(S)$

Heuristic: $h(S)$

Result (a, s), Result (b, s) \rightarrow Result (c, s)

strip: strips

op (Action: Go there), precond: At home / At home (m), effect: At there / At there (m)

Unit: U

Unit (a, b) \rightarrow Unit (c, d)

Unit (a, b) \rightarrow Unit (c, d)

Append CS to KB

KB: knowledge base

Partial plan: $\langle A, partial \rangle$

Complete plan: $\langle A, complete \rangle$

Goal: G

Initial state: S_0

Successor function: $SUC(S, a)$

Path cost: $c(a)$

Goal test: $G(S)$

Heuristic: $h(S)$

Speed, cost select subgoal

choose operator, plan operators, speed, cost

Reserve threats: plan

Unit (a, b) \rightarrow Unit (c, d)

Unit (a, b) \rightarrow Unit (c, d)

Goal: G

Initial state: S_0

Successor function: $SUC(S, a)$

Path cost: $c(a)$

Goal test: $G(S)$

Heuristic: $h(S)$

Reserve threats: plan

Speed, cost select subgoal

choose operator, plan operators, speed, cost

Unit (a, b) \rightarrow Unit (c, d)

Unit (a, b) \rightarrow Unit (c, d)

Goal: G

Initial state: S_0

Successor function: $SUC(S, a)$

Path cost: $c(a)$

Goal test: $G(S)$

Heuristic: $h(S)$

- ① $\frac{\partial L}{\partial x} = 0$
- ② $\frac{\partial L}{\partial y} = 0$
- ③ $\frac{\partial L}{\partial z} = 0$
- ④ $\frac{\partial L}{\partial \lambda} = 0$
- ⑤ $\frac{\partial L}{\partial \mu} = 0$
- ⑥ $\frac{\partial L}{\partial \nu} = 0$
- ⑦ $\frac{\partial L}{\partial \omega} = 0$
- ⑧ $\frac{\partial L}{\partial \phi} = 0$
- ⑨ $\frac{\partial L}{\partial \psi} = 0$
- ⑩ $\frac{\partial L}{\partial \chi} = 0$
- ⑪ $\frac{\partial L}{\partial \eta} = 0$
- ⑫ $\frac{\partial L}{\partial \theta} = 0$
- ⑬ $\frac{\partial L}{\partial \rho} = 0$
- ⑭ $\frac{\partial L}{\partial \sigma} = 0$
- ⑮ $\frac{\partial L}{\partial \tau} = 0$
- ⑯ $\frac{\partial L}{\partial \kappa} = 0$
- ⑰ $\frac{\partial L}{\partial \lambda} = 0$
- ⑱ $\frac{\partial L}{\partial \mu} = 0$
- ⑲ $\frac{\partial L}{\partial \nu} = 0$
- ⑳ $\frac{\partial L}{\partial \omega} = 0$
- ㉑ $\frac{\partial L}{\partial \phi} = 0$
- ㉒ $\frac{\partial L}{\partial \psi} = 0$
- ㉓ $\frac{\partial L}{\partial \chi} = 0$
- ㉔ $\frac{\partial L}{\partial \eta} = 0$
- ㉕ $\frac{\partial L}{\partial \theta} = 0$
- ㉖ $\frac{\partial L}{\partial \rho} = 0$
- ㉗ $\frac{\partial L}{\partial \sigma} = 0$
- ㉘ $\frac{\partial L}{\partial \tau} = 0$
- ㉙ $\frac{\partial L}{\partial \kappa} = 0$
- ㉚ $\frac{\partial L}{\partial \lambda} = 0$
- ㉛ $\frac{\partial L}{\partial \mu} = 0$
- ㉜ $\frac{\partial L}{\partial \nu} = 0$
- ㉝ $\frac{\partial L}{\partial \omega} = 0$
- ㉞ $\frac{\partial L}{\partial \phi} = 0$
- ㉟ $\frac{\partial L}{\partial \psi} = 0$
- ㊱ $\frac{\partial L}{\partial \chi} = 0$
- ㊲ $\frac{\partial L}{\partial \eta} = 0$
- ㊳ $\frac{\partial L}{\partial \theta} = 0$
- ㊴ $\frac{\partial L}{\partial \rho} = 0$
- ㊵ $\frac{\partial L}{\partial \sigma} = 0$
- ㊶ $\frac{\partial L}{\partial \tau} = 0$
- ㊷ $\frac{\partial L}{\partial \kappa} = 0$
- ㊸ $\frac{\partial L}{\partial \lambda} = 0$
- ㊹ $\frac{\partial L}{\partial \mu} = 0$
- ㊺ $\frac{\partial L}{\partial \nu} = 0$
- ㊻ $\frac{\partial L}{\partial \omega} = 0$
- ㊼ $\frac{\partial L}{\partial \phi} = 0$
- ㊽ $\frac{\partial L}{\partial \psi} = 0$
- ㊾ $\frac{\partial L}{\partial \chi} = 0$
- ㊿ $\frac{\partial L}{\partial \eta} = 0$

Goal 60: 7.5g

Goal 61: 7.5g

Goal 62: 7.5g

Goal 63: 7.5g

Goal 64: 7.5g

Goal 65: 7.5g

Goal 66: 7.5g

Goal 67: 7.5g

Goal 68: 7.5g

Goal 69: 7.5g

Goal 70: 7.5g

① [X1/1, 21/11]

② [X1/1, 21/11]

③ [X1/1, 21/11]

④ [X1/1, 21/11]

⑤ [X1/1, 21/11]

⑥ [X1/1, 21/11]

⑦ [X1/1, 21/11]

⑧ [X1/1, 21/11]

⑨ [X1/1, 21/11]

⑩ [X1/1, 21/11]

$$\frac{\partial R}{\partial x} = \beta \quad \frac{\partial R}{\partial x} = 2x \quad \frac{\partial R}{\partial x} = (A+AT)x$$

Boosting:

$$W_i^{(t)} = W_i^{(1)} x e^{-d_i} \quad (21)$$

$$Z_j = \frac{1}{\sum_i W_i^{(t)} x e^{-d_i}}$$

计算每个节点的权重

Z_j 是归一化后的权重

$$\frac{1}{N} \sum_j W_j I(G_j(x) = y)$$

$$\sum_j W_j I(G_j(x) = y)$$

Final: Speed \rightarrow Speed (as a limit)

Speed $<$ Speed (limit)

Speed $<$ Speed (Finish)

(Speed by effect (m/c))

Final: Speed \rightarrow Speed (as a limit)
Speed $<$ Speed (limit)
Speed $<$ Speed (Finish)
(Speed by effect (m/c))



扫描全能王 创建

1. 贪心 (Divide and Conquer)
 2. DP: 动态规划 (Top-down / Bottom-up)
 3. 递归 (Recursion)
 4. 排序 (Sorting)
 5. 字符串 (String)

1. activity selectn.
 2. task scheduln.
 3. Huffman coding

4. 排序: $O(n \log n)$
 快速排序 (Quick Sort)
 递归: $O(n \log n)$
 非递归: $O(n)$

5. 字符串匹配: KMP algorithm
 时间复杂度: $O(n)$
 空间复杂度: $O(n)$

6. 字符串: 字符串匹配, 字符串操作
 字符串: 字符串匹配, 字符串操作

7. 字符串: 字符串匹配, 字符串操作
 字符串: 字符串匹配, 字符串操作

① 数组 $a[100]$ $a[100][100]$
 ② 栈: `vector` `size()`
`push_back()`
`pop_back()`

③ 队列: `queue`
`push()`
`pop()`
`front()`
`back()`

④ 堆: `priority_queue`
`top()`
`pop()`
`push()`
`empty()`

二叉树
 中序遍历
 前序遍历
 后序遍历

二叉树遍历
 前序遍历
 中序遍历
 后序遍历

Huffman 树
 以 n 个字符为叶子节点
 构造 2 个节点

⑤ 堆排序
 时间复杂度: $O(n \log n)$
 空间复杂度: $O(1)$

二分查找
`low = 1`
`high = L`
`while (low <= high)`
`mid = (low + high) / 2`
`if (==) return mid`
`else if (kval < arr[mid]) high = mid - 1`
`else low = mid + 1`

平衡树 (AVL)
 时间复杂度: $O(\log n)$
 空间复杂度: $O(n)$

set
`set<int> s`
`s.insert(x)`
`s.erase(x)`
`s.find(x)`
`s.lower_bound(x)`
`s.upper_bound(x)`

priority_queue
`priority_queue<int, vector<int>, greater<int>>`
`push()`
`pop()`
`top()`
`size()`

⑥ 图论: dijstra
 最短路径
 ⑦ 图论: 并查集
 判断是否连通

hash
`hash<string>`
`hash-map<string, int>`
`string hash = "123"`