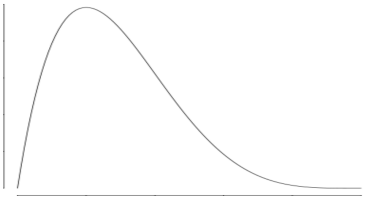
Aughdon Breslin
Professor Jie Shen

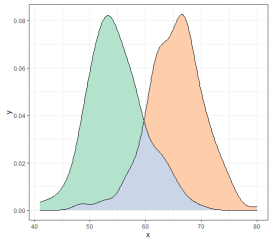
10/18/2022 Lecture 6

Measuring Discrimination in Human Decisions

- Discrimination in the Law
    - Disparate Treatment:
        - Individuals are treated differently due to prejudice against racial, gender, and protected trait groups.
    - **Unjustified Disparate Impact** (our focus)
        - A policy that appears neutral delivers differential results that cannot be justified by a valid, nondiscriminatory interest.
- Discrimination in Economics
    - Taste-based Discrimination
        - The decision maker shows a willingness to discriminate at the expense of utility
    - Statistical Discrimination
        - To maximize profit, the decision-maker draws logical conclusions based on group membership.
- Taste-based Discrimination
    - Decision makers derive utility from discriminating and thus act sub-optimally [relative to a profit-maximizing agent]
        - Example: A mother hires a less-qualified female nanny over a more-qualified male nanny to satisfy the employer's gender bias.
- Testing for Discrimination
    - Applying the same standard to all individuals is optimal [e.g., hiring everyone above a certain threshold, regardless of group membership]
    - Statistical discrimination tests in human decisions often aim to determine whether decision-makers apply different standards based on discrimination
- A motivating example
    - Police need probable cause to search a vehicle for contraband

        - Do officers apply the same probable cause standard equally to drivers of all races? [If not, they could find more contraband while conducting the same number of searches.]
        - Sacrifices optimacy to satisfy personal bias
- Benchmark Test
    - A simple test for discrimination - Are white and black drivers searched at similar rates?
        - A higher search rate for black drivers might indicate a lower (and discriminatory) bar for searching them
        - But without more information, it could also be the case that whites and blacks are held to the same standard, but that more black drivers are above the search threshold.
- Outcome Test
    - A "better" test for discrimination - Are the search outcomes the same for drivers of different races? [i.e., are the hit rates equal?]
        - A lower hit rate for black drivers might mean they are searched on the basis of less evidence.
- Risk Distributions
    - Likelihood of Possessing Contraband from [0,1]





    - Different average risk = different rates of carrying contraband
    - Different standard deviation = harder to tell rate of carrying contraband
- Problem with the Outcome Test
    - Police search if there's a greater than 50% chance they'll find contraband. [A facially neutral threshold].
    - If one group has more guilty people, the hit rate will be higher, which incorrectly suggests bias against a group with the same average of say 0.3, but with a lower variance
- Problem with Infra-Marginality
    - Infra marginal statistics:

        - Those that average over individuals away from the margin
        - Depends on both the threshold applied and the distribution of risk.
    - These statistics are imperfect proxies for the threshold, and hence problematic measures of taste-based discrimination.
- Infra-Marginality in Raleigh, NC
    - 16% Black Hit Rate vs 13% White Hit Rate
    - Searches of black drivers are more successful than searches of white drivers, and so the outcome test suggests bias against white drivers.
    - Black drivers in Raleigh are three times more likely to carry contraband in plain view, and so their risk distribution has a very heavy tail.
        - Tests for discrimination that account for the shape of the risk distributions find that officers apply a lower standard when searching black people.
        - Infra-marginality is real. In this case, the outcome test failed to detect bias against black drivers.
- Connection to ML Predictions
    - We can think of the hit rate as the precision of human decision makers. It is the fraction of those classified positive (i.e., searched) who had contraband.
    - As the Raleigh example shows, precision can be a misleading proxy for the threshold that is applied, and thus it is a problematic measure of discrimination.
- Algorithmic Risk Assessments
    - Many high-stakes decisions are made by first estimating the risk of an individual based on the available information.
    - Lending is based on the risk of default; pretrial detention is based on the risk of pretrial recidivism.
        - Recidivism: the tendency of a convicted criminal to reoffend.
    - Decisions guided by statistical risk assessments can be more effective and fair than those made by intuition alone.
- Pretrial Detention - A detailed case study
    - Judges must decide which arrested defendants should be released while awaiting trial and which should be detained.

    - The goal is to balance the social and financial costs of incarceration with the benefits of reducing pretrial crime.
- Risk Assessments in Broward County, FL
    - ProPublica analyzed 3,000 white and black defendants assigned COMPAS scores in Broward County, Florida. [Also determined whether these defendants recidivated.]
- A risk assessment tool
    - The Public Safety Assessment [New Jersey, San Francisco, elsewhere]
    - Weighted Checklist, (i.e. Violent offense? 20 years or younger? Pending charge at time of offense? Prior conviction? Prior violent conviction(s)?)
- Key Assumptions - Common to most papers on algorithmic fairness
    - We know the true label Y (i.e., whether a defendant would have reoffended if released).
        - [Y is true counterfactual, with no measurement error.]
    - We know the true risk: $r_x = P(Y = 1|X = x)$
- From Features to Decisions
    - Features → Prediction → Risk → Decision → Detain? Y/N
    - How should we go from features to decisions?
        - The risk $r_x = P(Y = 1|X = x)$ is fixed once we choose the features X
- Risk Distributions
    - Likelihood of violent recidivism is a right-skewed distribution from [0,1]
    - The mean is fixed for all choices of X. [The base rate of recidivism]
        - The shape can change based on our choice of X
- From Risk to Decisions
    - It's common to use a threshold on risk [The decision is independent of the features given risk.]
- Choosing a Threshold
    - A threshold of t means that we're willing to detain at most $\frac{1}{t}$ people.
- Applying a Threshold
    - A threshold rule optimally trades off between detention and recidivism.
- Taste-based Discrimination
    - We could detain fewer members of a smaller-variance group of the same mean while decreasing overall detention and crime

- Fairness of a Single Threshold
    - Equally risky people are treated equally, regardless of group membership.
    - No taste-based discrimination.
    - Inline with legal norms. This is what is done in practice
- Popular Mathematical Definitions of Fairness
    - Calibration
        - Outcome is independent of group membership given risk.
    - Classification Parity
        - False positive rates are equal across groups.
    - Anti-classification
        - Protected characteristics are not used by the algorithm.
    - All three definitions are problematic formalizations of long-standing legal and social norms.
        - 1. Calibration does not preclude taste-based discrimination
        - 2. Classification parity almost always leads to taste-based discrimination
        - 3. Anti-classification often leads to taste-based discrimination
- Calibration
    - Conditional on risk score groups should reoffend at equal rates
    - Calibration does not preclude taste-based discrimination
        - preclude: prevent from happening
- Discrimination with Calibrated Scores
    - Detain defendants with $r_x > 0.5$
    - Probability of violent recidivism [0,1].
    - A distributed group with mean = 0.4 would be detained while a uniformly 0.4 group would not
- False Positive Rate Parity
    - The false positive rates are equal for all groups
    - False positive rate = $\frac{wouldn't\ have\ reoffended\ \&\ were\ detained}{wouldn't\ have\ reoffended}$
    - Error Rate Disparities in Broward County
        - 31% of black defendants who did not reoffend vs. 15% of white defendants who did not reoffend were deemed high risk of committing a violent crime

- Higher false positive rates for black defendants

- False Positive Rate Misconceptions
  - 1. A higher false positive rate for some group implies discrimination
    - A higher risk group receiving the same standard would yield a higher false positive rate
    - This is because the false positive rate is an infra-marginal statistic
  - 2. A higher false positive rate for a minority gorup is due to a lack of data, either:
    - a) a lack of training examples [rows]
      - If the base rates differ, the risk distributions will always differ, regardless of how many data points we have. And even if the base rates are similar, the distributions may still differ.
    - b) a lack of predictive features [columns]
      - if we acquire new predictive features the risk distribution shifts outwards, since we're better able to distinguish between recidivists and non-recidivists.
      - This can actually increase the false positive rate, since it might result in more defendants lying above the threshold.
      - Equalizing FPRs by Ignoring Information
        - The average black defendant is below the threshold.
        - As the risk scores get worse we lose the ability to distinguish between high and low risk defendants.
        - False positive rates are lower when we lose the ability to distinguish between high and low risk defendants.
        - False positive rates are equalized when the black risk scores have almost no predictive validity
  - 3. False positive Rates are a proxy for group well-being
    - Fairness using Confusion Matrices

	Non-recidivist	Recidivist
Released	TN	FN
Detained	FP	TP

        - All these definitions compare infra-marginal statistics
- Is the data biased?
    - Biased predictors
        - Features that are differentially predictive
    - Biased labels
        - Y doesn't perfectly measure what we care about
- Biased Predictors
    - Marijuana arrests are likely biased: minority users are more likely to be arrested than white users
    - Including it in the model will overstate its correlation
- Problem with Anti-classification
    - Gender neutral risk models can lead to taste-biased discrimination
    - One can fix this problem by using one model for men and another for women [or by including gender in the model]
- Biased Labels
    - In reality we measure who is arrested or convicted, not who [would have] committed the crime
    - Increased policing in minority areas might make certain arrest types [e.g., for drugs] a problematic measure of actual crime
    - Some outcomes [e.g., violent crime] seem less prone to measurement error.

10/4/2022 Lecture 5

Statistical Machine Learning

- Gradient Descent
    - Consider minimization without constraints:

      $min_w F(w),\ w \in R^d$
    - Gradient Descent:
        - 1. Initialize w^0 arbitrarily, e.g. $w^0 = 0$
        - 2. For $t = 1, 2, \ldots$

          $w^t = w^{t-1} - \eta_t \nabla F(w^{t-1})$
    - Goal:
        - $w^t \to w^*$, where $w^* = arg\ min\ F(w)$

        - In few iterations (cheap computation)
- Smoothness
    - $F(w)$ is smooth if for any $w_1,\ w_2 \in R^d$

      $||\nabla F(w_2) - \nabla F(w_1)||_2 \leq L||w_2 - w_1||_2$

      $\frac{||\nabla F(w_2)-\nabla F(w_1)||_2}{||w_2-w_1||_2} \leq L$

      L must be positive.
    - In English:
        - If you move slightly along the function, the gradient should not change much, or
        - The gradient has a stable change when w changes.
    - Examples:
        - $F(w) = w^2,\ \nabla F = 2w,\ \frac{||2*(w_1-w_2)||}{||w_1-w_2||}$

          $ratio = 2,\ L = 2$

          Constant L -> Smooth!
        - $F(w) = |w|,\ w_1 = -\varepsilon,\ w_2 = \varepsilon,\ (\varepsilon > 0)$

          $\nabla F(w_1) = -1,\ \nabla F(w_2) = +1$

          $\frac{-1-1}{2\varepsilon} \leq L$

          $ratio = \frac{1}{\varepsilon} \leq L, \varepsilon \to 0$

          Since we cannot find a constant L, we know this function is not smooth around the origin. (w_1,w_2 cannot be bounded by a constant L)
        - $F(w) = w^4,\ w_1, w_2$

          $\frac{4||w_2^3-w_1^3||}{||w_2-w_1||} \leq L$

          $ratio = 4(w_1^2 + w_1w_2 + w_2^2) \leq L$

          Non-constant L -> non-smooth
    - Homework:
        - $F(w) = e^{-w}$
        - $F(w) = \frac{1}{1+e^{-w}}$
    - L is the maximum eigenvalue of the Hessian Matrix

- Hessian Matrix
  - Given $F(w),\ w \in R^d$,

    Hessian: $\nabla^2 F(w) \in d \times d$

    ratio: $\frac{||\nabla F(w_2) - \nabla F(w_1)||_2}{||w_2 - w_1||_2} \leq L$

    Think of $\nabla F(w_1)$ as $g(w_1)$ and same for w_2.
  - Mean-value: R^d

    $g(w_1) - g(w_2) = g'(w) * (w_1 - w_2)$

    $R^d \uparrow \qquad R^d \uparrow \qquad\qquad R^d \uparrow$

    $\nabla F(w_1) - \nabla F(w_2) = \nabla^2 F(w) * (w_1 - w_2)$

    $\exists \lambda \in [0,1],\ w = \lambda w_1 + (1 - \lambda) w_2$

    Let $v = w_1 - w_2$.

    ratio: $\frac{||\nabla F(w_1) - \nabla F(w_2)||_2}{||w_1 - w_2||_2} \leq L$

    $= \frac{||\nabla^2 F(w) * (w_1 - w_2)||}{||w_1 - w_2||} \leq L$

    $= \frac{||\nabla^2 F(w) * (w_1 - w_2)||}{||v||} \leq L$

    $max_{v \in R^d} \frac{||M * v||}{||v||} \rightarrow greatest\ eigenvalue$

    L is the greatest eigenvalue of $\nabla^2 F(w)$.
  - Smoothness is so important because it guarantees that Gradient Descent will decrease the function value with a sufficiently small learning rate η.
- Informal Analysis
  - $w^t = w^t - \eta \nabla F(w^t)$

    $F(w^{t+1}) - F(w^t)$

    $= F(w^t - \eta \nabla F(w^t)) - F(w^t)$

    $=< \nabla F(w), -\eta \nabla F(w^t) >$

    $= -\eta < \nabla F(w), \nabla F(w^t) >$
  - If $\eta \rightarrow 0$, the guesses are sufficiently close to w ($w \approx w^t,\ \nabla F(w) \approx \nabla F(w^t)$).

- Convergence Analysis
  - Suppose $F(w)$ is convex and L-smooth. Pick $0 < \eta \leq \frac{1}{L}$. Then for all $t \geq 1$ where t is the number of iterations in gradient descent,

    $$F(w^t) - F(w^*) \leq \frac{||w^0 - w^*||_2^2}{2\eta} * \frac{1}{t}$$

  - In particular, picking $\eta = \frac{1}{L}$ gives

    $$F(w^t) - F(w^*) \leq \frac{L||w^0 - w^*||_2^2}{2t}$$

- Implications
  - $F(w^t) - F(w^*)$ v.s. $||w^t - w^*||_2$

    $w^* \in argmin(F(w))$, since for many $F(w)$, there may be infinite w*
    - i.e. F(w)=1, all values of w are the minimum.
    - If w* is unique, then we can look at $||w^t - w^*||$.
- Iteration Complexity
  - Given $\varepsilon \in (0, 1)$, we want $F(w^t) - F(w^*) \leq \varepsilon$, $t \geq \frac{1}{\varepsilon}$ (ε is iteration complexity)
    - $F(w^t) - F(w^*) \leq \frac{1}{t}$
    - If gradient descent runs $\frac{1}{\varepsilon}$ times, we can guarantee $F(w^t) - F(w^*) \leq \varepsilon$ with learning rate $\eta = \frac{1}{L}$.
- Faster Rate of Convergence
  - Strongly Convex: For any $w_1, w_2 \in R^d$

    $$||\nabla F(w_2) - \nabla F(w_1)||_2 \geq \alpha||w_2 - w_1||_2$$

  - $\alpha = min\ eigenvalue\ of\ \nabla^2 F(w)$
  - Functions Satisfying Strongly Convex:
    - If you add $F(w) = w^2$ to any function, it will be Strongly Convex

      $F(w) = g(w) + ||w^2||^2$

      $convex \uparrow \quad \geq 0 \uparrow$

      $\nabla^2 F = \nabla^2 g + 2I \geq 2$
    - $12w^2$ is considered 12-strongly convex because it is strongly convex in a certain domain.
- Theoretical Guaranteed Convergence

- Suppose $F(w)$ is α-strongly convex and L-smooth. Let $\{w^t\}_{t\geq 1}$ be the iterates generated by GD where $0 < \eta \leq \frac{2}{\alpha+L}$. Then $\forall t \geq 1$,

  $$||w^t - w^*||_2 \leq \sqrt{1 - \frac{2\eta\alpha L}{\alpha+L}} * ||w^0 - w^*||_2$$

  - In particular picking $\eta = \frac{2}{\alpha+L}$ gives (where the condition number $c = \frac{L}{\alpha}$)

    $$||w^t - w^*||_2 \leq (1 - \frac{2}{c+1})||w^{t-1} - w^*||_2$$

  - Converges linearly/geometric rate of convergence
  - Typically the best one can hope for
- $\forall t \geq 1,\ ||w^t - w^*||_2 \leq (1 - \frac{2}{c+1})^t ||w^0 - w^*||_2 \leq e^{\frac{-2t}{c+1}} ||w^0 - w^*||_2$
- For any pre-defined error $0 < \epsilon < 1$,

  $t = c * log(\frac{||w^0 - w^*||_2}{\epsilon})$ is the iteration complexity, which is much smaller than simple convex functions

- Overall Computational Complexity

Condition	Guarantee	# of iterations
α-SC, L-smooth	$\|\|w^t - w^*\|\|_2 \leq \epsilon$	$clog(\frac{1}{\epsilon})$
L-smooth	$F(w^t) - F(w^*) \leq \epsilon$	$\frac{L}{\varepsilon}$

  - Gradient Descent solves linear regression efficiently: $d^2(n + d)$ v.s. $ndc * log(\frac{1}{\epsilon}$ where d is the number of dimensions.

Stochastic Gradient Descent

- How to Improve Gradient Descent wrt. computational cost?
  - Gradient Descent
    - Program: $min_w F(w),\ s.t.\ w \in R^d$
    - $GD \in O(nd)$
  - SGD Algorithm
    - Initialize w^0, say $w^0 = 0$
    - For $t = 1, 2, \ldots$

      Uniformly draw i_t from $\{1, 2, \ldots, n\}$ and update

$$w^t = w^{t-1} - \eta_t \nabla f_{i_t}(w^{t-1})$$

stochastic↑

- $E[w^t] = E[w^{t-1}]\eta_t E[\nabla f_{i_t}(w^{t-1})]$
  - $SGD \in O(d)$
  - $total\ time = \frac{cost}{iteration} * \#\ of\ iterations$

9/27/2022 Lecture 4

Probably Approximately Correct (PAC) Model

- D {X*Y}
- Hypothesis Class/Concept Class

  $H_i = \{h_w : x \to y\}$
  - Example: $y = sign(w * x)$

    Half-space: $h_w : x \to sign(w * x)$ maps input to label

    $h_w(x) = sign((w_1 * x_1)^2 + (w_2 * x_2)^2)$

    $x \epsilon R^d,\ x = (x_1, x_2),\ where\ x_1, x_2 \epsilon R^{\frac{d}{2}}$

    $w \epsilon R^d,\ x = (w_1, w_2),\ where\ w_1, w_2 \epsilon R^{\frac{d}{2}}$
- Intersections of Half-Spaces
  - All points above a curve w^* are positive (+), all below are negative (-)
  - For multiple curves, we can define all the points within region A as +, outside as -
    - Region(s) defined as + just depends on what model you're making
- For an $X: [0, 1],\ Y: \{+: x > \alpha,\ \ -: x \leq \alpha\}$,

  draw n samples, observe [-,+,+,-, ..., α_{pred}] until $|\alpha_{pred} - \alpha| \leq \varepsilon$.

  ε is commonly known as the error rate, δ is known as the probability of failure of achieving an $|\alpha_{pred} - \alpha| \leq \varepsilon$.

  We want the probability that our guess is wrong to be less than ε:

  $$P(h(x)_{guess} \neq y) \leq \varepsilon$$

Label Efficiency (Active Learning)

- Don't need to label all of the data

- Draw x, then decide if we want to draw y to label x
  x: Sample Complexity (samples drawn)
  y: Label Complexity (labels drawn)
- On a line from 0 to 1, there exist a partition of (-)'s somewhere on the left, and (+)'s somewhere on the left.
  - Binary Search: If we query in the middle, we can get a result, say (+), and then know that all entries to the right are +, but somewhere on the left there exists the partition.
  - Repeat the process at 0.25, and say find a minus, now knowing all entries to the left of 0.25 are (-). Repeat until we find the partition.
    $\Theta(log(n))$ label complexity.
  - TODO: 30min in
- $sign(wx) = sign(a_1 sign(w_1 x) + a_2 sign(w_2 x) + \ldots + a_d sign(w_d x))$

AdaBoost

- Given hypothesis class H but hypothesis $h_{guess} \notin H$, then its considered improper PAC learning
- Learning H may be very difficult/expensive (especially with discontinuous functions)
  - Result of H = step function may be a tuned sigmoid that does not belong to step functions

Empirical Risk Minimization

- For any hypothesis $h \epsilon H$,
  loss: $L(h; x, y),\ h(x) \leftrightarrow y$
  - Example: $h(x) = wx$
    $L() = (wx - y)^2$ TODO 45 min
    Draw $S = \{(x_i, y_i)\}$.
    Loss of our samples: $L_S = \frac{1}{n} \sum_{i=1}^{n} L(h; x_i, y_i)$
    We want the $min_{h \epsilon H} L_S(h) = h_{guess}$.
- How can we solve?
  - We want to optimize runtime
- Why does it make sense?
  - There exists a true function that outputs y
    $\exists h^*(x) = y$

- Our model should be close enough to the true model within epsilon

  $||h_{guess} - h^{*}|| \leq \varepsilon$
- We use statistical estimations so the number of samples will increase accuracy (and expense)

- $min_{w \epsilon C} F(w)$

  $C = \{w: ||w||_2 \leq \alpha\}$ where C is the Constraint set $C \epsilon R^d$ and α is some constant
  - Could also be defined in L_1 with $||w||_1$ or whatever other measuring norms

Gradient Descent

$w^{t+1} = w^t - \eta_t \nabla F(w^t)$

- When will gradient descent fail?
  - When it stops, but does not stop at the global optimum

    The weight stops changing:

    $t: \ w^{t+1} = w^t$

    The gradient reaches 0:

    $w^t - \eta_t \nabla F(w^t) = w^t$

    $\Rightarrow \eta_t \nabla F(w^t) = 0$
  - If the function is concave!
- Convex Set C
  - For all combinations of two points in the set, draw a line connecting them. If the line is contained within the set, the function is convex.

    $\forall u, v \in C, \ \forall \lambda \epsilon [0, 1]$

    $\lambda u + (1 - \lambda) v \in C$
    - A ball is convex, a sphere is not.
      - A ball is filled in while a sphere is just the outline.
  - A function is convex if and only if the region above its graph is a convex set.
    - This region is the function's epigraph.

      $epi(F) = \{(x, y): y \geq F(x)\}$
  - Assume existence of ∇F.

    $\forall x, y \in domain \ of \ F$

    $F(y) \geq F(x) + < \nabla F(x), y - x >$

- Example: $F(x) = x_1^2 + x_2 + x_3$

  $\frac{\delta F}{\delta x_1} = 2x_1,\ \frac{\delta F}{\delta x_2} = 1,\ \frac{\delta F}{\delta x_3} = 1$

  $\rightarrow \nabla F = (2x_1, 1, 1)$

- Assume existence of $\nabla^2 F$. This is called a Hessian matrix.
  - $\nabla^2 F$ means second order derivative of F(x)
  - $\nabla F = (\frac{\delta F}{\delta x_1}, \frac{\delta F}{\delta x_2}, \frac{\delta F}{\delta x_3})$ each partial is a real value
  - Gradient $g(x) \epsilon R$, take gradient with respect to every value
  - Example: $F(x) = x_1^2 + x_2 + x_3$

    $\nabla F(x) = (2x_1, 1, 1)$

    $\nabla^2 F(x) = \frac{\delta F}{\delta x} = [2\ 0\ 0]$

    $[0\ 0\ 0]$

    $[0\ 0\ 0]$
- A positive semi-definite matrix M exists if all eigenvalues of M are ≥ 0.

  $\Rightarrow M \geq 0$

  $\forall v \epsilon R^d,\ v^T M v \geq 0 \Rightarrow M \geq 0$
- $F(x)\ is\ convex \Leftrightarrow \nabla^2 F(x) \geq 0,\ \forall x \in domain$
  - Example: $F(x) = x_1^2 x_2 + x_3$

    $\frac{\partial F}{\partial x_1}$ TODO 1:40

- GD is good for convex functions!
  - $min(F(w))\ such\ that\ w \in C.$
    - 1. F(w) must be convex.
    - 2. C must be convex.
- How soon will gradient descent succeed? What value should the total # of iterations T be?
  - $T = \infty,\ 100,\ 10^6$?
  - $T = f(w^0, \eta, \varepsilon)$
  - Goal: $||\widehat{w} - w^*|| \leq \varepsilon$
  - Example: $F(w) = \frac{1}{2} w^2$
    - Guess that $w^0 = 100$.

- $\nabla F(w) = w$
- Let $\eta = 100,\ (\eta\ must\ be\ > 0)$.
- $w^{t+1} = w^t - 100w^t = -99w^t$

  $w^1 = -9900$

  $w^2 = 99 * 9900$

  $w^3 = -99^2 * 9900$

  $w^t = (-1)^t(99)^t * 100$

  - Diverges, maybe we should choose a more conservative learning rate.
- Let $\eta = \frac{1}{2},\ (\eta\ must\ be\ > 0)$.
- $w^{t+1} = w^t - \frac{1}{2}w^t = \frac{1}{2}w^t$

  $1 \to \frac{1}{2} \to \frac{1}{4} \to \ ... \ \to 0$

  - Converges with linear learning rate

- Example 2: $F(w) = |w|$

  $\nabla F(w) = \{+1 : w > 0;\ \ -1 : w \leq 0\}$

  $w^{t+1} = w^t - \eta sign(w^t)$

  $Let\ w^0 = 1,\ \eta = \frac{1}{2}$.

  $w^1 = 1 - \frac{1}{2}(+1) = \frac{1}{2}$

  $w^2 = \frac{1}{2} - \frac{1}{2}(+1) = 0$

9/20/2022 Lecture 3

Classification

- Binary Classification
  - $X \epsilon R^d\ on\ interval\ [a, b]$
  - Y: $\{+1, -1\}$
  - D: $X \times Y$
  - Sample data belonging to the population

    $(x_1, y_1) \ ... \ (x_n, y_n) \sim_{i.i.d} D$

    ^ given ^ ^ unknown

        - x_i is an image, y_i is a label
    - $h\colon x \to y$
    - $L(h; x, y) = l\{h(x) \neq y\}$
    - Error Rate of h (err(h)):

      $E[L(h; x, y)] = 1 * P(h(x) \neq y) + 0 * P(h(x) = y) = P(h(x) \neq y)$
    - Goal: Find h such that for any given target error rate,

      $\varepsilon \in (0, 1),\ err(h) \leq \varepsilon\ w.p.\ 1 - \delta,\ \delta \in (0, 1)$, we can find a lower and upper data sample
        - $P(h(x) \neq y) \leq \varepsilon$
    - Draw the data $(x_1, y_1) \ldots (x_n, y_n) \sim D$ to try to find the distribution
- Probably Approximately Correct Learning, Valiant 1985 Example
    - $X\colon [0, 1],\ Y\colon \{+1, -1\}$
    - True Model: $Y = \{+1\colon x \geq \alpha, -1\colon x < \alpha\}$
        - If our guess is between $[\alpha, 1]$, the output is +1

          If our guess is between $[0, \alpha]$, the output is -1
    - Goal: Find α_{pred} such that $|\alpha - \alpha_{pred}| \leq \varepsilon$.
    - Distribution D on X is uniform
    - Draw data samples to get an upper and lower bounds
        - For example, if we get $(\alpha_{low}, -1),\ (\alpha_{high}, +1)$, then the true α lies within $[\alpha_{low}, \alpha_{high}]$. We can update $\alpha_{pred} = \frac{\alpha_{low} + \alpha_{high}}{2}$.
        - We need data samples that fall within $[\alpha_{low}, \alpha]$ and $[\alpha, \alpha_{high}]$ to get bounds
    - If we find data samples within $[\alpha_-, \alpha_+],\ where\ \alpha_{+,-} = \alpha \pm \varepsilon$, we're done!

      $Event\ 1\colon [\alpha_-, \alpha],\ Event\ 2\colon [\alpha, \alpha_+]\ w.p.\ 1 - \delta$
    - How likely is it for a sample to fall within $[\alpha_-, \alpha]$? Let's say the size of the interval is $\frac{\varepsilon}{2}$.
        - $z_i = \{1\colon [\alpha_-, \alpha],\ 0\colon otherwise\}$
        - $P(z_i = 1) = \frac{\varepsilon}{2}$
        - $E[z_i] = \frac{2}{\varepsilon}$

          You'd expect to draw $\frac{2}{\varepsilon}$ times to get one sample that falls within $[\alpha_-, \alpha]$.

- $S = \sum_{i=1}^{n} z_i \geq 1$
  - $P(z_i) = \{1: \frac{\varepsilon}{2}, 0: otherwise\}$
  - $E[S] = \frac{\varepsilon}{2} n$
- $P(|S - E[S]| \geq t) \leq 2e^{-\frac{t^2}{n(b-a)^2}}$

  We can use Hoeffding's Inequality to find the likelihood w.p. $1 - \frac{\delta}{2}$ that one of the events will happen (Event 1).

  $RHS \rightarrow 2e^{-\frac{t^2}{n}} \leq \frac{\delta}{2}$

  Divide by 2 and take the inverse

  $\rightarrow e^{\frac{t^2}{n}} \geq \frac{4}{\delta}$

  Take the log of both sides

  $\rightarrow \frac{t^2}{n} \geq log(\frac{4}{\delta})$
- With probability $1 - \frac{\delta}{2}$, the compliment, $|S - E[S]| \leq t$, will happen:

  $|S - E[S]| \leq t$

  $\Rightarrow S \geq E[S] - t = \frac{\varepsilon}{2} n - t$

  We want to make sure RHS ≥ 1. Let $t = \frac{\varepsilon n}{4}$.

  $S \geq \frac{\varepsilon n}{2} - \frac{\varepsilon n}{4} = \frac{\varepsilon n}{2} \geq 1$

  Then solve for n:

  $n \geq \frac{2}{\varepsilon}$

  We now have a sample size constraint.
- Bringing back $\frac{t^2}{n} \geq log(\frac{4}{\delta})$, we can plug in $t = \frac{\varepsilon n}{4}$.

  $\frac{(\frac{\varepsilon n}{4})^2}{n} \geq log(\frac{4}{\delta})$

  $\frac{\varepsilon^2 n}{16} \geq log(\frac{2}{\delta})$

  Solve for n.

  $n \geq \frac{16}{\varepsilon^2} log(\frac{2}{\delta})$

  Let $n = \frac{16}{\varepsilon^2} log(\frac{2}{\delta})$ to ensure that w.p. $1 - \frac{\delta}{2}$, we'll get one point that's within $[\alpha\_, \alpha]$

- Since it's symmetric, we can make the same conclusion for $[\alpha, \alpha_+]$.
  - This yields $2n$ as the number of required drawings w.p. $1 - \frac{\delta}{2}$.

- Develop an Algorithm without just drawing points
  - Adaboost

    $X \epsilon R^d,\ Y = \{0, 1\}$
    - Combining weak learners to make better model
    - We have k weak learners $h_1, \ldots, h_k$.
      - $err(h_i) \leq \frac{1}{2} - \gamma,\ \gamma \epsilon (0, \frac{1}{2})$
      - $Let\ \gamma = 0.01,\ err(h_i) \leq 0.49$
    - Main idea: For each prediction function, we predict correctly with a noticeable portion (say 51%), and incorrectly the rest. For all of the incorrectly labeled data, we can make it more precise. We start with baseline weight on all predictions, notice that all of them successfully label the image. We can de-emphasize this example in the next iteration. Conversely, we can emphasize majority-incorrectly-labeled images to ensure the model improves in the next iteration.

      $(x_i, y_i) \ldots (x_n, y_n)$ where all weights start at $\frac{1}{n}$
      - If w_i is large: hard example for $h_1, \ldots, h_k$
      - If w_i is tiny: easy example for $h_1, \ldots, h_k$
      - Loss in boosting is weighted
        - If loss approaches 0, we are done with that example. If loss is large, we need to keep training on it
    - $SVM\ Loss$: $\frac{1}{n} \sum_{i=1}^{n} hinge(w; x_i, y_i)$
    - $Linear\ Reg$: $\frac{1}{n} \sum_{i=1}^{n}$ TODO
    - $Boosting\ Loss$: TODO

- Algorithm:
  - $t = 1, \ldots, T$
  - $L_i^t = |h_t(x_i) - y_i|$
    - i[th] sample (x_i, y_i)
  - $w_i^{t+1} = w_i^t * \beta^{L_i^t}$
  - $h_{t+1} = min \sum_{i=1}^{n} w_i^{t+1} loss(x_i, y_i)$
  - $T: h_1 \ldots h_T$
  - $Given\ x,\ h_1(x),\ \ldots,\ h_T(x)$

9/13/2022 Lecture 2

Process of Building an AI Model to recognize face

- Download Images
  - Need to label the images as +1 or -1
  - Need to make sure the labels are accurate
    - Have 50 people label the same image, use Alg, return one label that is hopefully correct
      - Workers [x_1, ..., x_n] can label -1 or +1
      - Lets say correct label is +1. S = Σx_i >= 1
    - Simple Alg = majority vote
      - Assume 60% of workers are accurate -> P(x = +1) = .6, P(x = -1) = .4 . How many workers do we need to ensure the majority vote is correct with a probability >0.99?

        $2e^{-t^2/2n} = 0.01\ (Allowable\ error)$

        $E[X]n - t = 1 \text{ --> } 0.2n - t = 1$
      - By Hoeffding's system of equations, we need at least 245 workers.
  - Goal: With a probability of 0.99, all labels are correct
    - IM1: 250 people -> correct with a probability of 0.99
    - IM2: 250 people -> correct w.p. of 0.99
      - P(IM1 & IM2) = 0.99 * 0.99 = 0.9801

        - P(IM1 & IM2 & … & IMn) = $(0.99)^n$ ~= 0
    - #workers/img grows with complexity O(logn) to maintain overall probability goals
    - Later, we'll see a way to make this O(1)
- Process Data
- Techniques to Make the Algorithm Work
- Result (+1 = face, -1 = no face)

Online Learning (Hedge)

- n players, 0 1 2 … n.
- Each has a different probability of winning the game (1v1 type game)
- Goal: Pick a strategy to find the best player
- L_i^t = the i^{th} player on their t^{th} iteration of the game had result L belonging to [0,1] (did not lose, lost)
    - Subscript = player, superscript = iteration
- $w_i^{t+1} = w_i^t * \beta^{L_i^t}$
    - Weights of all players start at 1/N, and change based on their result
    - β falls within (0,1)
    - $L_i^t = 0\ (player\ won),\ L_i^t = 1\ (player\ lost)$
    - If the player wins, their weight does not change. If they lose, their weight diminishes
- Loss at t^{th} iteration = $\langle w^t, L^t \rangle$

$$E_{i \sim w^t}[L_i] = L_1 w_1 + L_2 w_2 + \ldots + L_N w_N = \sum_{i=1}^{N} w_i L_i$$

- If you can only observe 1 player's feedback, this is called bandit's classification
- Loss of Algorithm (of all experts across all iterations): L_A, normalized with 1/sum(w_i)

$$L_A = \frac{1}{\sum_{i=1}^{N} w_i^t} \sum_{t=1}^{T} \left(\sum_{i=1}^{N} w_i^t L_i^t \right)$$

- Loss of Expert: L_i

$$L_i = L_i^1 + L_i^2 + \ldots + L_i^T = \sum_{t=1}^{T} L_i^t$$

- Loss of the algorithm is bounded by any loss of expert i plus

  $L_A \leq \ min_{1\leq i\leq N} L_i + \ log(N)$

  $\frac{L_A}{T} \leq \ min(\frac{L_i}{T}) \ + \frac{log(N)}{T} \rightarrow \ 0 \ as \ T \rightarrow \ \infty$

- Want to prove these bounds

  $L_i \leq \sum_{i=1}^{N} w_i^{t+1} \leq L_A$

  - For $\sum_{i=1}^{N} w_i^{t+1} \geq L_i$:

    $\sum_{i=1}^{N} w_i^{t+1} \geq w_i^{t+1} = w_i^{t} * \beta^{L_i^t} = w_i^{t-1} * \beta^{L_i^t} * \beta^{L_i^{t-1}} = \ ...$

    $= \ w_i^{1} * \beta^{L_i^t} * \beta^{L_i^{t-1}} *... * \beta^{L_i^1} = w_i^{1} * \beta^{L_i^t + L_i^{t-1} + ... + L_i^1}$

    $= w_i^{1} * \beta^{L_i} = \frac{1}{N}\beta^{L_i}$

    - Each weight is defined by the previous weight*beta; simplify beta multiplication
    - Why does this prove the lower bound?

  - For $\sum_{i=1}^{N} w_i^{t+1} \leq L_A$, we'll need to use this inequality:

    Proven elsewhere: $\alpha^r \leq 1 - (1 - \alpha)r \ for \ all \ \alpha \geq 0, \ r\epsilon[0,1]$

    $\sum_{i=1}^{N} w_i^{t+1} = \sum_{i=1}^{N} w_i^{t}\beta^{L_i^t}$

    Think of beta as alpha, and L_i^t as r, and plug into inequality

    $RHS \rightarrow \sum_{i=1}^{N} w_i^{t}\beta^{L_i^t} \leq \sum_{i=1}^{N} w_i^{t}(1 - (1 - \beta)L_i^t)$

    Multiply and divide by sum of w_i^t.

    $= \sum_{i=1}^{N} w_i^{t} * \frac{1}{\sum_{i=1}^{N} w_i^t} * \sum_{i=1}^{N} w_i^{t}(1 - (1 - \beta)L_i^t)$

    Now for each w_i^t, we divide by sum of w_i^t.

    $= \sum_{i=1}^{N} w_i^{t} * \sum_{i=1}^{N} \frac{w_i^t}{\sum_{i=1}^{N} w_i^t}(1 - (1 - \beta)L_i^t)$

Let $S = \sum_{i=1}^{N} {w_i}^t$ and distribute.

$$= \sum_{i=1}^{N} {w_i}^t * \sum_{i=1}^{N} (\frac{{w_i}^t}{S} - (1 - \beta) * \frac{{w_i}^t}{S} * {L_i}^t)$$

We can then bring the summations in.

$$= \sum_{i=1}^{N} {w_i}^t * (\sum_{i=1}^{N} \frac{{w_i}^t}{S} - \sum_{i=1}^{N} (1 - \beta) * \frac{{w_i}^t}{S} * {L_i}^t)$$

Sum of $w_i^t/S = 1$, and pull 1-beta out.

$$= \sum_{i=1}^{N} {w_i}^t * (1 - (1 - \beta) * \sum_{i=1}^{N} \frac{{w_i}^t}{S} * {L_i}^t)$$

Sum of $(w_i^t/S)*L_i^t$ is the loss of the algorithm at generation t (L_A^t).

$$= \sum_{i=1}^{N} {w_i}^t * (1 - (1 - \beta) * {L_A}^t)$$

Bringing back the LHS, we get this inequality

$$\sum_{i=1}^{N} {w_i}^{t+1} \leq \sum_{i=1}^{N} {w_i}^t * (1 - (1 - \beta) * {L_A}^t)$$

We can use this to continue the RHS, replacing the sum of (w_i^t/S) with t-1

$$\leq \sum_{i=1}^{N} {w_i}^{t-1} * (1 - (1 - \beta) * {L_A}^{t-1}) * (1 - (1 - \beta) * {L_A}^t)$$

Continue this down to w_i^1. sum of w_i^1 is equivalent to 1

$$\leq \sum_{i=1}^{N} {w_i}^1 * \prod_{t=1}^{T} (1 - (1 - \beta) * {L_A}^t) = 1 * \prod_{t=1}^{T} (1 - (1 - \beta) * {L_A}^t)$$

$$LHS \leq \prod_{t=1}^{T} (1 - (1 - \beta){L_A}^t) = \prod_{t=1}^{T} (1 + (\beta - 1){L_A}^t)$$

We'll now need to use another inequality, plug in beta stuff for x

Proven elsewhere: $1 + x \leq e^x \; for \; all \; x \geq 0$

$$\prod_{t=1}^{T} (1 + (\beta - 1){L_A}^t) \leq \prod_{j=1}^{t} e^{(\beta-1){L_A}^j}$$

Can simplify to be addition of exponent rather than multiplication of base

$$RHS = e^{(\beta-1)\sum_{j=1}^{t} {L_A}^j} = e^{(\beta-1)L_A}$$

- Now bring back the lower bound to compare against the upper bound

$\frac{1}{N} * \beta^{L_i} \leq e^{-(1-\beta)L_A}$

Invert both sides

$e^{(1-\beta)L_A} \leq \frac{N}{\beta^{L_i}}$

Take the log of both sides

$(1 - \beta)L_A \leq log(N) + L_i * log(\frac{1}{\beta})$

Divide by (1-beta)

$L_A \leq \frac{log(\frac{1}{\beta})}{1-\beta} * L_i + \frac{log(N)}{1-\beta}$

Let $\beta = 1/2$.

$L_A \leq \frac{log(\frac{1}{1/2})}{1-\frac{1}{2}} * L_i + \frac{log(N)}{1-\frac{1}{2}}$

$L_A \leq \frac{log(2)}{\frac{1}{2}} * L_i + \frac{log(N)}{\frac{1}{2}}$

$L_A \leq 2log(2) * L_i + 2log(N)$ (over T iterations)

Divide both sides by T

$\frac{L_A}{T} \leq 2log(2) * \frac{L_i}{T} + \frac{2log(N)}{T}$

$\frac{2log(N)}{T} \to 0 \; as \; T \; increases.$

Initially, there's going to be some gap between your prediction and the best expert's prediction, but after many iterations, we will agree with each other, which ensures that the average loss goes to 0.

9/6/2022 Lecture 1

Syllabus

- Review of Calculus, Probability, Linear Algebra
- Random Projection — Unsupervised
- Singular value decomposition, principal component analysis
- K-Means Clustering, Subspace Clustering
- Dictionary Learning and Sparse Coding
- Low-rank Matrix Estimation, with applications to recommender systems
- Computational Social Science

- Robust Mean Estimation, Robust Classification
- Algorithmic Fairness

Books

- Understanding Machine Learning From Theory to Algorithms

Linear Algebra Overview

- Vector Properties
- Matrix Properties
- (1) A: n x d, y: n x 1, Ax = y, x = ?
    - When is this feasible?
    - How to find x? Is it unique?
        - Unique when n ≥ d, A is invertible, $A^TAx = A^Ty$, x is sparse (most elements are zero)

Probability Overview

- Expected Value
    - Discrete: $E[X] = \sum_{i=1}^{n} x_i P(x_i)$
    - Continuous: $E[X] = \int_0^{\infty} xp(x)dx$ where p(x) is the pdf of x
    - Practice: Play a game for money. P(X = 1) = 0.6, P(X = -1) = 0.4
        - $E[X] = -1 * 0.4 + 1 * 0.6 = 0.2$
            - We're expected to win $0.20 every time we play.
            - When can we win $100? 0.2n = 100 -> n = 500
                - $S = 100,\ E[S] = 0.2n = 100\ when\ n = 500$
        - However, we will not always win exactly $100 on our 500th play every time.
        - When can we win $100 w.p. 0.99?
- Markov's Inequality
    - If x > 0, then for all t> 0,

      $P(x \geq t) \leq \frac{E[x]}{t}$
        - Proof of correctness
            - $E[x] = \int_0^{\infty} xp(x)dx$

              p(x) is the probability density function of x on [0,inf]

$$= \int_0^t xp(x)dx + \int_t^\infty xp(x)dx \geq 0 + \int_t^\infty tp(x)dx$$

[0,t] integral is ≥ 0, and, in [t, inf] integral, x ≥ t, so LHS is ≥ RHS,

$$RHS = t\int_t^\infty p(x)dx = t * P(x \geq t)$$

t doesn't depend on x, so can be pulled out, and [t,inf] integral can be substituted for the probability that x is ≥ t.

$$E[x] \geq t * P(x \geq t)$$

Summarize what we've stated so far.

$$P(x \geq t) \leq \frac{E[x]}{t}$$

Divide by t to get the original equation.

- Proof of tightness
  - Tightness: $x^2 \geq 0$ is tight because when x = 0, $x^2 = 0$
    - For this reason, $x^2 + 1$ is not tight because when x = 0, $x^2 + 1 = 1$
  - Prove $P(x \geq t * E[x]) \leq \frac{1}{t}$ to show tightness.

    $Let\ P(X = 0) = 1 - \frac{1}{t},\ P(X = 1) = \frac{1}{t}$

    $E[X] = 0 * (1 - \frac{1}{t}) + 1 * \frac{1}{t} = \frac{1}{t}$

    $P(x \geq t * E[x]) \rightarrow P(x \geq t * \frac{1}{t}) \rightarrow P(x \geq 1) = \frac{1}{t}$

  - Negative random variables
    - moment-generating function

- Set E[X]/t = 0.01

  -> t = 100*E[X]

  -> P(x ≥ 100*E[X]) <= 0.01

  -> P(x < 100*E[X] ≥ 0.99

  -> 0 < x <= 100*E[X]

- Variance
  - $Var(X) = E[X - E[X]]^2 = E[X^2] - E[X]^2$
  - $Var[x_i] = E[x_i^2] - E[x_i]^2$
- Chebyshev's Inequality
  - $P(|X - E[X]| > t) <= Var(X)/t^2$

- Markov's: $P(Y > t^2) <= E[Y]/t^2$
  -> $Y = |X - E[X]|^2$
  -> $P(Y > t^2)$ -> $P(Y^{1/2} > t)$
  -> $P(|X - E[X]| > t) = E[|X - E[X]|^2]/t^2$
  -> $P(|X - E[X]| > t) = Var[X]/t^2$
- Chebyshev is more powerful, but also needs to know the random variable's variance

- Moment
  - E[X] is the first moment, $E[X^2]$ is the second, $E[X^3]$ is the third, and so on.
  - Moment Generating Function
    - Way to specify its probability distribution
    - $E[e^{\lambda X}]$ and $E[e^{\lambda Y}]$
    - Same moments ⇔ same distribution
- Hoeffding's Inequality
  - Symmetric Bernoulli Distribution: P(x = 1) = P(x = -1) = ½
  - Theorem: Let $x_1, x_2, ..., x_n$ be independent symmetric Bernoulli random variables. Let a = (a_1, a_2, ..., a_n) belongs to R^n. Then for any t ≥ 0,
    $P(\sum_{i=1}^{n} x_i \geq t) \leq e^{\frac{-t^2}{2n}}$
  - Proof of correctness
    - $P(\sum_{i=1}^{n} x_i \geq t) \leq e^{\frac{-t^2}{2n}}$

      $LHS = P(\lambda * \sum_{i=1}^{n} x_i \geq \lambda * t) \rightarrow P(e^{\lambda * \sum_{i=1}^{n} x_i} \geq e^{\lambda * t}) \leq_{by\ Markov's} \frac{E[e^{\lambda * \sum_{i=1}^{n} x_i}]}{e^{\lambda * t}}$

      - $numerator = E[e^{\lambda * \sum_{i=1}^{n} x_i}]$
        $= E[\prod_{i=1}^{n} e^{\lambda * x_i}]$
        Works because $e^{a+b+c+d} = e^a e^b e^c e^d$
        $= \prod_{i=1}^{n} E[e^{\lambda * x_i}]$
        We can pull out the product because all of the random variables x_i are independent.

$$= \prod_{i=1}^{n} (e^{\lambda} * \frac{1}{2} + e^{-\lambda} * \frac{1}{2})$$

Use Expected Value formula with Symmetric Bernoulli Distribution so each $x_i \in [-1, 1]$ has P(x) = ½. Also, this is where the Moment Generating Function gets used.

$$= \prod_{i=1}^{n} (\frac{e^{\lambda}+e^{-\lambda}}{2}) \leq_{by\ a\ different\ proof} \prod_{i=1}^{n} (e^{\frac{\lambda^2}{2}})$$

$$= e^{\frac{n\lambda^2}{2}}$$

Product doesn't depend on i, so just gets multiplied together n times (which is, of course, represented as base^n).

- $LHS = P(\sum_{i=1}^{n} x_i \geq t) \leq_{by\ Markov's} \frac{E[e^{\lambda * \sum_{i=1}^{n} x_i}]}{e^{\lambda * t}} \leq_{by\ above} \frac{e^{\frac{n\lambda^2}{2}}}{e^{\lambda t}}$ for all λ>0

  $P(\sum_{i=1}^{n} x_i \geq t) \leq e^{\frac{n\lambda^2}{2} - \lambda t}$ for all λ>0. Now how can we pick λ?

  $Let\ \lambda = \frac{t}{n} \rightarrow P(\sum_{i=1}^{n} x_i \geq t) \leq e^{\frac{n(\frac{t}{n})^2}{2} - (\frac{t}{n})t}$

  $\rightarrow P(\sum_{i=1}^{n} x_i \geq t) \leq e^{\frac{t^2}{2n} - \frac{t^2}{n}} \rightarrow P(\sum_{i=1}^{n} x_i \geq t) \leq e^{-\frac{t^2}{2n}}$

- Generalize to non-symmetric distribution
  - $Let\ a_i \leq x_i \leq b_i,\ and\ S = \sum_{i=1}^{n} x_i$

    $P(|S - E[S]| \geq t) \leq 2e^{-\frac{2t^2}{\sum_{i=1}^{n} (b_i - a_i)^2}}$

    - If we assume all the a_i's and b_i's are the same, we can simplify.

      $P(|S - E[S]| \geq t) \leq 2e^{-\frac{2t^2}{n(b-a)^2}}$

    - RHS is known as the Failure Probability
  - We can use this to further solve the game from last week.

    The game: $P(x_i = 1) = 0.6$, $P(x_i = -1) = 0.4$. How many rolls to have $S \geq 100$? How many rolls to have $S \geq 100$ w.p. ≥ 0.99?

    $a = -1,\ b = 1$

$$P(|S - E[S]| \geq t) \leq 2e^{-\frac{2t^2}{n(b-a)^2}}$$

$$\rightarrow P(|S - E[S]| \geq t) \leq 2e^{-\frac{2t^2}{n(1-(-1))^2}}$$

$$\rightarrow P(|S - E[S]| \geq t) \leq 2e^{-\frac{2t^2}{n*2^2}}$$

$$\rightarrow P(|S - E[S]| \geq t) \leq 2e^{-\frac{t^2}{2n}}$$

$$\Leftrightarrow P(|S - E[S]| \leq t) \geq 1 - 2e^{-\frac{t^2}{2n}}$$

(1) $2e^{-\frac{t^2}{2n}} = 0.01$

(2) $0.2n - t = 100$ 2 variables, 2 equations -> solve for n

$n \approx 1019,\ t \approx 104$

We need to roll 1019 times to have a probability of winning $100 with a probability of success at 0.99.