**Stat 100a, Introduction to Probability.**

Outline for the day:

1. Submit hw2 immediately to statgrader@stat.ucla.edu .
2. Sample ch5 problems.
3. Harman and Negreanu running it twice.
4. Project teams.
5. Continuous random variables and density.
6. Uniform random variables.
7. Exponential distribution.
8. Normal distribution.
9. Pareto.

Read through chapter 6.

Midterm 2 is next Tue. I will email you instructions at 955am again.

http://www.stat.ucla.edu/~frederic/100A/S20

♠ ♣ ♥ ♦

1. Submit hw2.
2. ch5 problems.

Let X = the # of hands until your 1st pair of black aces. What are E(X) and SD(X)?

X is geometric(p), where p = 1/C(52,2) = 1/1326.
E(X) = 1/p = 1326.
SD = ($\sqrt{q}$) / p, where q = 1325/1326. SD = 1325.5.

What is P(X = 12)?
$q^{11}p$ = 0.0748%.

You play 100 hands. Let X = the # of hands where you have 2 black aces. What is E(X)? What is P(X = 4)?
X is binomial(100,p), where p = 1/1326.
E(X) = np = .0754.
P(X = 4) = C(100,4) $p^4$ $q^{96}$ = .000118%.

**3. Harman and Negreanu, and running it twice.**

Harman has 10♠ 7♠ . Negreanu has K♥ Q♥ . The flop is 10♦ 7♣ K♦ .

Harman's all-in. $156,100 pot. P(Negreanu wins) = 28.69%. P(Harman wins) = 71.31%.

Let X = amount Harman has after the hand.

If they run it once, E(X) = $0 x 29% + $156,100 x 71.31% = **$111,314.90.**

<u>If they run it twice, what is E(X)?</u>

There's some probability $p_1$ that Harman wins both times ==> X = $156,100.

There's some probability $p_2$ that they each win one ==> X = $78,050.

There's some probability $p_3$ that Negreanu wins both ==> X = $0.

E(X) = $156,100 x $p_1$ + $78,050 x $p_2$ + $0 x $p_3$.

If the different runs were *independent*, then $p_1$ = P(Harman wins 1st run & 2nd run) would = P(Harman wins 1st run) x P(Harman wins 2nd run) = 71.31% x 71.31% ~ 50.85%.

But, they're not quite independent! Very hard to compute $p_1$ and $p_2$.

*However, you don't need $p_1$ and $p_2$ !*

X = the amount Harman gets from the 1st run + amount she gets from 2nd run, so

E(X) = E(amount Harman gets from 1st run) + E(amount she gets from 2nd run)

= $78,050 x P(Harman wins 1st run) + $0 x P(Harman loses first run)

+ $78,050 x P(Harman wins 2nd run) + $0 x P(Harman loses 2nd run)

= $78,050 x 71.31% + $0 x 28.69% + $78,050 x 71.31% + $0 x 28.69% = **$111,314.90.**

HAND RECAP Harman 10♠ 7♠ Negreanu K♥ Q♥ The flop is 10♦ 7♣ K♦ .

Harman's all-in. \$156,100 pot.P(Negreanu wins) = 28.69%. P(Harman wins) = 71.31%.

---------------------------------------------------------------------------

The standard deviation (SD) changes a lot! **Say they run it once.** (see p127.)

$V(X) = E(X^2) - \mu^2$.

$\mu$ = \$111,314.9, so $\mu^2$ ~ \$12.3 billion.

$E(X^2)$ = (\$156,100$^2$)(71.31%) + ($0^2$)(28.69%) = \$17.3 billion.

V(X) = \$17.3 billion - \$12.3 bill. = \$5.09 billion. SD $\sigma$ = sqrt(\$5.09 billion)~\$71,400.

So if they run it once, Harman expects to get back about \$111,314.9 **+/- \$71,400**.

**If they run it twice?** Hard to compute, but approximately, if each run were

independent, then $V(X_1+X_2) = V(X_1) + V(X_2)$,

so if $X_1$ = amount she gets back on 1st run, and $X_2$ = amount she gets from 2nd run,

then $V(X_1+X_2) \sim V(X_1) + V(X_2)$ ~ \$1.25 billion + \$1.25 billion = \$2.5 billion,

The standard deviation $\sigma$ = sqrt(\$2.5 billion) ~ \$50,000.

So if they run it twice, Harman expects to get back about \$111,314.9 **+/- \$50,000.**

4. Project teams.

Last time I assigned you to project teams. The project is problem 8.2, page 249.

At the end of class today, if you didn't get a project team last time, stick around and I will assign you.

You need to write code to go all in or fold.

```
bruin = function (numattable, cards, board, round, currentbet, mychips, pot, roundbets,
      blinds, chips, ind, dealer, tablesleft) {
## all in with any pair higher than 7s, or if lower card is J or higher,
## or if you have less than 3 times the big blind
a = 0
if ((cards[1, 1] == cards[2, 1]) && (cards[1, 1] > 6.5)) a = mychips
if (cards[2,1] > 10.5) a = mychips
if(chips < 3*blinds) a = mychips
a
} ## end of bruin
```

cards[1,1] is your higher card (2-14).

cards[2,1] is your lower card (2-14).

cards[1,2] and cards[2,2] are suits of your higher card & lower card.

Optional. In R, try

```
install.packages("holdem")
library(holdem)
library(help="holdem")
```

gravity, timemachine, tommy, ursula, vera, william, and xena are examples.

```
help(tommy)
```

```
tommy = function (numattable, cards, board, round, currentbet, mychips,
   pot, roundbets, blinds, chips, ind, dealer, tablesleft)
{ a = 0
   if (cards[1, 1] == cards[2, 1])
      a = mychips
   a
}
```

It might say a1 instead of a, and crds1 instead of cards. I sometimes name variables with 1 at the end so the names do not conflict with other R names.

```
help(vera)
# All in with a pair, any suited cards, or if the smaller card is at least 9.
vera = function (numattable1, crds1, board1, round1, currentbet,
    mychips1, pot1, roundbets, blinds1, chips1, ind1, dealer1,
    tablesleft) {
  a1 = 0
  if ((crds1[1, 1] == crds1[2, 1]) || (crds1[1, 2] == crds1[2,2]) ||
    (crds1[2, 1] > 8.5)) a1 = mychips1
  a1
}
```

You need to email me your function, to frederic@stat.ucla.edu, by Sun Jul26, 8pm. It should be written (or cut and pasted) simply into the body of the email. If you write it in Word, save as text first, and then paste it into the email. Just submit one email per team.

Here is bruin again.

```
bruin = function (numattable, cards, board, round, currentbet, mychips, pot, roundbets, blinds, chips, ind, dealer,
        tablesleft) {
## all in with any pair higher than 7s, or if lower card is J or higher, or if you have less than 3 times the big blind
a = 0
if ((cards[1, 1] == cards[2, 1]) && (cards[1, 1] > 6.5)) a = mychips
if (cards[2,1] > 10.5) a = mychips
if(chips < 3*blinds) a = mychips
a
} ## end of bruin
```

The top 3 finishers get 13,8,5 points respectively. 1000 chips. Blinds start at 10/20 and increase by 50% every 10 hands.
numattable = number of players at your table.
cards[1,1] = number of higher card, cards[2,1] = lower card. cards[1,2] and cards[2,2] are their suits.
currentbet = the maximum bet so far this betting round.
mychips is how many chips you have left.
pot = the number of chips in the pot.
blinds = the amount of the big blind. The small blind will be half as much. Both are rounded to integers.
chips is a vector indicating how many chips everyone at your table has left.
ind is your seat number at the table.
dealer is the seat number of the dealer at your table.
tablesleft is how many tables of up to 10 players each are left in the tournament.
Ignore board, round, and roundbets because you have to be all in or fold before the flop.

```
Iveybruin = function (numattable, cards, board, round, currentbet,mychips,
      pot, roundbets, blinds, chips, ind, dealer,tablesleft) {
## all in with any 9 pair or higher, or if lower card is 10 or higher,
## or if I have less than 3 times the big blind
a = 0
if ((cards[1, 1] == cards[2, 1]) && (cards[1, 1] > 8.5)) a = mychips
if (cards[2,1] > 9.5) a = mychips
if(mychips < 3*blinds) a = mychips
a
} ## end of Iveybruin
```

```
acj = function(numattable, cards, board, round, currentbet, mychips, pot, roundbets, blinds, chips, ind, dealer,
tablesleft){
        # all in with pocket pair of Queens, Kings, or Aces, or AK of any suits
        # all in with AQ, AJ, A10, KQ if same suit and no one else is all in yet
        # all in with any pocket pair if only 1-2 players left to play and nobody is all in yet
        # all in if chip count less than 3 times the big blind with any cards
        a = 0
        if((cards[1,1] == cards[2,1]) && (cards[1,1] >= 12)) a = mychips
        if((cards[1,1] == 14) && (cards[2,1] == 13)) a = mychips
        if(currentbet <= blinds){
                if((cards[1,1] == 14) && (cards[2,1] >= 10) && (cards[1,2] == cards[2,2])) a = mychips
                if((cards[1,1] == 13) && (cards[2,1] == 12) && (cards[1,2] == cards[2,2]))a = mychips
                }
        big.blind = dealer + 2
        if(big.blind > numattable) big.blind <- big.blind - numattable
        players.left = big.blind - ind
        if(players.left < 0) players.left = players.left + numattable
```

```
acj = function(numattable, cards, board, round, currentbet, mychips, pot, roundbets, blinds, chips, ind, dealer,
tablesleft){
# all in with any pocket pair if only 1-2 players left to play and nobody is all in yet
# all in if chip count less than 3 times the big blind with any cards
                    if(currentbet <= blinds){
                              if((players.left <= 2) && (cards[1,1] == cards[2,1])) a = mychips
                    }
                    if(mychips < 3 * blinds) a = mychips
          a
}
```

If you did not get a team last time, I will sort you into teams at the end of class using breakout rooms.

**5. Continuous random variables and their densities, ch6.1.**

Density (or pdf = Probability Density Function) $f(y)$:

$$\int_B f(y)\, dy = P(X \text{ in } B).$$

If $F(c)$ is the cumulative distribution function, i.e. $F(c) = P(X \leq c)$,

then $f(c) = F'(c)$.

The survivor function is $S(c) = P(X > c) = 1 - F(c)$.

If $X$ is a continuous rv, then $P(X \leq a) = P(X < a)$, because $P(X = a) = \int_a^a f(y)dy = 0$.

Expected value, $\mu = E(X) = \int y\, f(y)\, dy$. (= $\sum y\, P(y)$ for discrete X.)

For any function g, $E(g(X)) = \int g(y)\, f(y)\, dy$. For instance $E(X^2) = \int y^2 f(y)dy$.

Variance, $\sigma^2 = V(X) = Var(X) = E(X-\mu)^2 = E(X^2) - \mu^2$.

$SD(X) = \sqrt{V(X)}$.

For examples of pictures of pdfs, see p104, 106, and 107.

**6. Uniform distribution.**

Recall for a continuous random variable X,
the pdf f(y) is a function where $\int_a^b f(y)dy = P\{X \text{ is in } (a,b)\}$,
$E(X) = \mu = \int_{-\infty}^{\infty} y\, f(y)dy$,
and $\sigma^2 = Var(X) = E(X^2) - \mu^2$. $sd(X) = \sigma$.
If X is uniform(a,b), then f(y) = 1/(b-a) for y in (a,b), and y = 0 otherwise.

For example, suppose X and Y are independent uniform random variables on (0,1), and Z = *min*(X,Y). **a)** Find the pdf of Z. **b)** Find E(Z). **c)** Find SD(Z).

**a.** For c in (0,1), $P(Z > c) = P(X > c \;\&\; Y > c) = P(X > c)\, P(Y > c) = (1-c)^2 = 1 - 2c + c^2$.
So, $P(Z \le c) = 1 - (1 - 2c + c^2) = 2c - c^2$.
Thus, $\int_0^c f(c)dc = 2c - c^2$. So f(c) = the derivative of $2c - c^2 = 2 - 2c$, for c in (0,1).
Obviously, f(c) = 0 for all other c.
**b.** $E(Z) = \int_{-\infty}^{\infty} y\, f(y)dy = \int_0^1 c\,(2-2c)\, dc = \int_0^1 2c - 2c^2\, dc = c^2 - 2c^3/3]_{c=0}^{1}$
$= 1 - 2/3 - (0 - 0) = 1/3$.
**c.** $E(Z^2) = \int_{-\infty}^{\infty} y^2 f(y)dy = \int_0^1 c^2 (2-2c)\, dc = \int_0^1 2c^2 - 2c^3\, dc = 2c^3/3 - 2c^4/4]_{c=0}^{1}$
$= 2/3 - 1/2 - (0 - 0) = 1/6$.
So, $\sigma^2 = Var(Z) = E(Z^2) - [E(Z)]^2 = 1/6 - (1/3)^2 = 1/18$.
$SD(Z) = \sigma = \sqrt{(1/18)} \sim 0.2357$.

## 7. Exponential distribution, ch 6.4.

Useful for modeling waiting times til something happens (like the geometric).

pdf of an exponential random variable is $f(y) = \lambda\, exp(- \lambda\, y)$, for $y \geq 0$, and $f(y) = 0$ otherwise.
The cdf is $F(y) = 1 - exp(- \lambda\, y)$, for $y \geq 0$.
If $X$ is exponential with parameter $\lambda$, then $E(X) = SD(X) = 1/\lambda$

If the total numbers of events in any disjoint time spans are independent, then these totals are Poisson random variables. If in addition the events are occurring at a constant rate $\lambda$, then the times between events, or *interevent times*, are exponential random variables with mean $1/\lambda$.

**Example.** Suppose you play 20 hands an hour, with each hand lasting exactly 3 minutes, and let $X$ be the time in hours until the end of the first hand in which you are dealt pocket aces. Use the exponential distribution to approximate $P(X \leq 2)$ and compare with the exact solution using the geometric distribution.

**Answer.** Each hand takes 1/20 hours, and the probability of being dealt pocket aces on a particular hand is 1/221, so the rate $\lambda$ = 1 in 221 hands = 1/(221/20) hours ~ 0.0905 per hour.
Using the exponential model, $P(X \le 2 \text{ hours}) = 1 - \exp(-2\lambda) \sim 16.556\%$.
This is an approximation, however, since by assumption $X$ is not continuous but must be an integer multiple of 3 minutes.
Let $Y$ = the number of hands you play until you are dealt pocket aces. Using the geometric distribution, $P(X \le 2 \text{ hours}) = P(Y \le 40 \text{ hands})$
$= 1 - (220/221)^{40} \sim 16.590\%$.

The survivor function for an exponential random variable is particularly simple: $P(X > c) = \int_c^\infty f(y)dy = \int_c^\infty \lambda \exp(-\lambda\, y)dy = -\exp(-\lambda\, y)]_c^\infty = \exp(-\lambda\, c)$.

Like geometric random variables, exponential random variables have the *memorylessness* property: if $X$ is exponential, then for any non-negative values $a$ and $b$, $P(X > a+b \mid X > a) = P(X > b)$. *(See p115).*
Thus, with an exponential or geometric random variable, if after a certain time you still have not observed the event you are waiting for, then the distribution of the *future,* additional waiting time until you observe the event is the same as the distribution of the *unconditional* time to observe the event to begin with.

**8. Normal distribution.**

So far we have seen two continuous random variables, the uniform and the exponential.

Normal. pp 115-117. mean = $\mu$, SD = $\sigma$, $f(y) = 1/\sqrt{(2\pi\sigma^2)}\ e^{-(y-\mu)^2/2\sigma^2}$.

Symmetric around $\mu$,

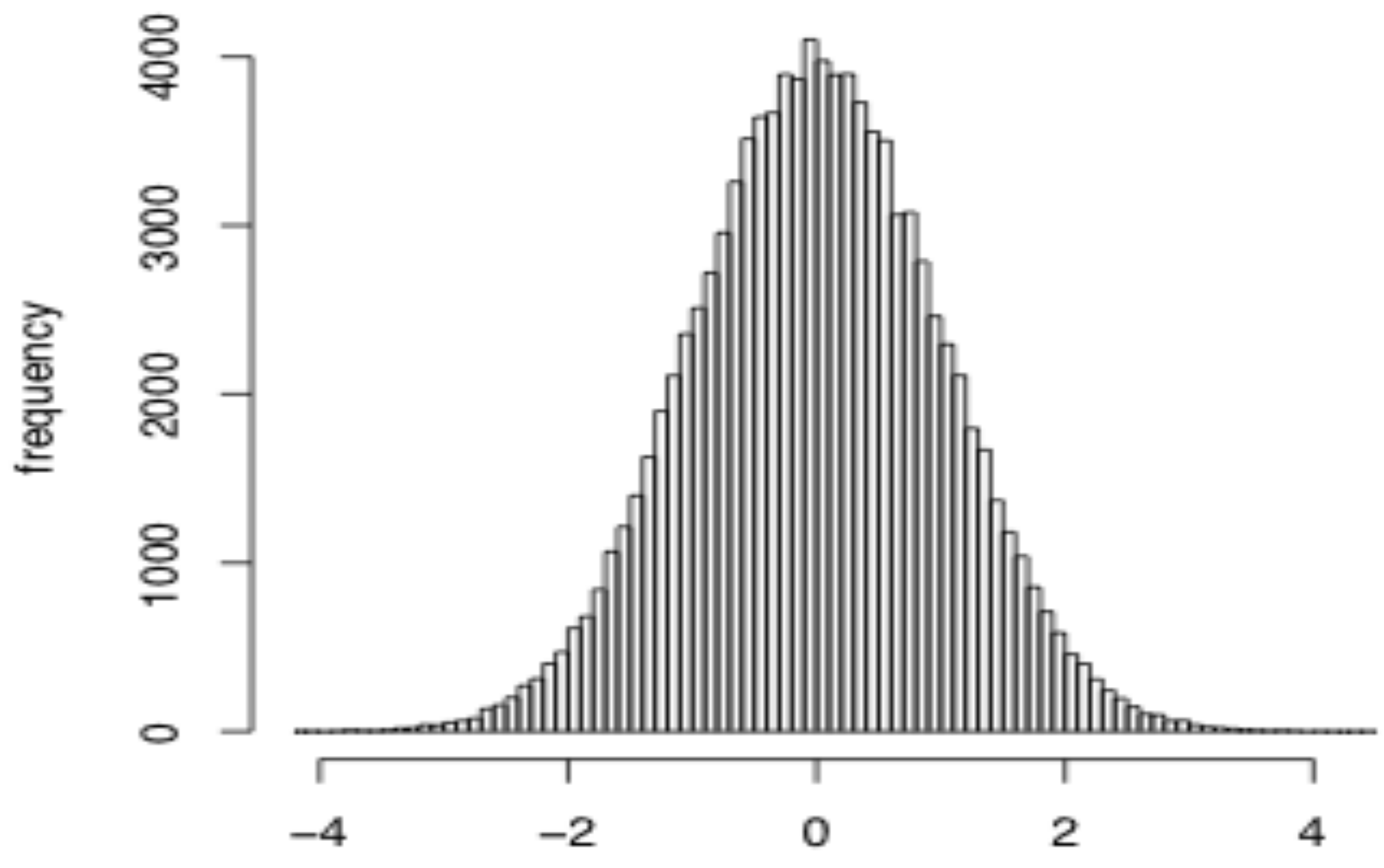
50% of the values are within 0.674 SDs of $\mu$,

68.27% of the values are within 1 SD of $\mu$, and

95% are within 1.96 SDs of $\mu$.

* Standard Normal. Normal with $\mu = 0$, $\sigma = 1$. See pp 117-118.

Standard normal density:
68.27% between -1.0 and 1.0
95% between -1.96 and 1.96



## 9. Pareto distribution, ch 6.6.

Pareto random variables are an example of *heavy-tailed* random variables, which means they have very, very large outliers much more frequently than other distributions like the normal or exponential.

For a Pareto random variable, the pdf is $f(y) = (b/a)\,(a/y)^{b+1}$, and the cdf is $F(y) = 1 - (a/y)^b$,

for $y>a$, where $a>0$ is the *lower truncation point*, and $b>0$ is a parameter called the *fractal dimension*.

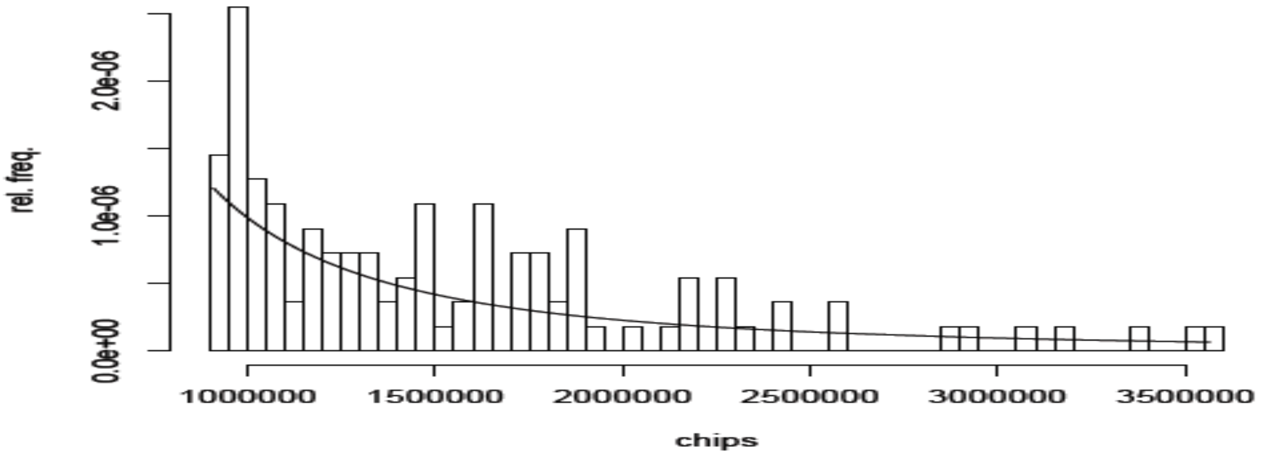



Figure 6.6.1: Relative frequency histogram of the chip counts of the leading 110 players in the 2010 WSOP main event after day 5. The curve is the Pareto density with $a = 900{,}000$ and $b = 1.11$.