Solution to In-Class Exercise

Problem: What is the expected value of a 5 card draw from a standard deck. Face cards are value 10, aces are value 11.

Solution:
Analytically solving this problem is difficult, as you need to condition each subsequent draw on the results of the previous draw. Let's start by determining the expected value of a single card from a fresh deck:

$$E[D1] = \frac{4(2+3+4+5+6+7+8+9+10+10+10+10+11)}{52}$$

Now, let's think about what the expected value of the second draw would look like. Because we removed one card from consideration on the first draw, we would need to express this. The way we could do this is by considering the expected value of the second draw conditional on each possible first draw.

$$E[D2|D1 = X] = (4(2+3+4+5+6+7+8+9+10+10+10+10+11) - X)/51$$

However, this requires us to know what the first card draw is. If we wanted to solve this to obtain the marginal conditional $E[D2]$ we need to take a sum across all values of $D1$

$$E[D2] = \sum_{X \in D} \frac{E[D2|D1]}{52}$$

Now, for $D3$, this gets even more complicated, as we now need to remove from consideration both $D1$ and $D2$, but $D2$ is dependent on $D1$.

$$E[D3|D2 = Y, D1 = X] = \frac{(4(2+3+4+5+6+7+8+9+10+10+10+10+11) - (X+Y))}{50}$$

$$E[D3] = \sum_{X \in D} \sum_{Y \in D-X} \frac{E[D3|D1 = X, D2 = Y]}{52 * 51}$$

In the above equation, the second summand is modified by which card we drew in the first draw. $Y \in D2 - X$ means here (and this is non-standard notation), that Y is from the deck D without card X.

Now, we can continue this logic for $D4$ and $D5$. I'll leave this as an exercise for the reader. But, this nested loop summand structure is a nightmare to calculate by hand, so here is the code that will get all the expected values for each draw.

```
deck = rep(c(2,3,4,5,6,7,8,9,10,10,10,10,11), 4)
d1_e = 0
for(i in 1:52){
  d1_e = d1_e+deck[i]/52
}
d2_e = 0
for(i in 1:52){
  deck_d2 = deck[-i]
  for(j in 1:51){
    d2_e = d2_e + deck_d2[j]/(52*51)
  }
}

d3_e = 0
for(i in 1:52){
  deck_d2 = deck[-i]
  for(j in 1:51){
    deck_d3 = deck_d2[-j]
    for(k in 1:50){
      d3_e = d3_e + deck_d3[k]/(52*51*50)
    }
  }
}

d4_e = 0
for(i in 1:52){
  deck_d2 = deck[-i]
  for(j in 1:51){
    deck_d3 = deck_d2[-j]
    for(k in 1:50){
      deck_d4 = deck_d3[-k]
      for(q in 1:49){
        d4_e = d4_e + deck_d4[q]/(52*51*50*49)
      }
    }
  }
}

d5_e = 0
for(i in 1:52){
  deck_d2 = deck[-i]
  for(j in 1:51){
    deck_d3 = deck_d2[-j]
    for(k in 1:50){
      deck_d4 = deck_d3[-k]
      for(q in 1:49){
```

```
        deck_d5 = deck_d4[-q]
        for(l in 1:48){
          d5_e = d5_e + deck_d5[l]/(52*51*50*49*48)
        }
      }
    }
  }
}

total_expected_value = d1_e+d2_e+d3_e+d4_e+d5_e
```

The analytic expected value is 36.538461544449.

Now, how does this compare to the expected value of 5 draws from fresh decks? All that would be is $5 * E[D1] = 36.5384615384615$.

These two values are different, but only after the 8th decimal point. Also, this shows that the expected value of a real 5 card hand is actually very very very slightly more than the approximated expected value of 5 cards from a fresh deck.

Now, we could also approximate this using a simulation study:

```
d1 = 0
d2 = 0
for(i in 1:100000){
  d1 = d1+ sum(sample(deck, 5, replace = F))/100000
  d2 = d2+ sum(sample(deck, 5, replace = T))/100000
}
```

Where d1 is the draw from the deck without replacement, while d2 is the draw from the deck with replacement. The numbers will change slightly each time you run this code, but can give you a reasonable sense of the outcomes. I got for d1 36.499 and for d2 36.496.

Again, these are approximations to the true expected value, you would need to use an infinite loop to truly converge on the values. Given that the real difference is so small between the true expected value of the 5 card hand and the expected value of the approximation, you would likely need to sample a truly ungodly number of iterations to converge to the point where you'd see the difference.

In theory, I'd conclude that the expected value of a 5 card draw from 1 deck is smaller than the expected value of a 5 card draw from 5 fresh decks. In practice, the expected values are, for all intents and purposes, identical, and the approximation is a reasonable one.

Part 2: What is the probability that you'd draw a hand with a value of 30 or greater?

I'm not even going to lay out the analytics for this, we are going to go straight into simulation and approximation.

To simulate, we just need to modify our code a tad:

```
d1 = vector()
d2 = vector()
for(i in 1:100000){
  d1[i] =  sum(sample(deck, 5, replace = F))
  d2[i] = sum(sample(deck, 5, replace = T))
}
mean(1*(d1> 30))
mean(1*(d2> 30))
```

For probabilities, I got .826 and .818 respectively. Given sampling variance, these are effectively identical.

But now, instead of approximating using simulation, let's calculate the variance of a true 5 card draw and use that and the expected value to approximate the distribution by making a normal assumption.

Unfortunately, the variance here is annoying to directly calculate, so let's use the simulation to determine the variance. I got a variance of 39.26512.

So, now, let's approximate the distribution of the value of a 5 card hand by $N(36.538461544449, 39.26512)$

From this approximation, the probability that you'd draw a hand with value 30 or greater is $.8516$.

Now, is this a reasonable approximation? It really depends on why you need these probabilities. If you are running a casino, a .03 difference in probability might add up. What these results are telling me is that the normal approximation is within the same order of magnitude as the simulated results, but might slightly overestimate the chances of drawing a 30 value hand.