Spread Prediction Using an MLE Driven Simulation Model

NBA spread betting is a popular game in America. Each year, hundreds of millions of dollars are wagered on NBA spread bets in Nevada casinos. The game is simple to play. For a given upcoming game, players select the team they think will have the higher final score after a known handicap is applied to the final total. A winning bet of U dollars will net the bettor U, whereas a losing bet of U dollars will cost 1.1*U. The percentage of excess that the loser needs to pay is known as the vig, and it performs a similar function to the rake in poker. Casinos set the handicap, also known as the line, so that the amount wagered on both teams is roughly equal. This guarantees the casinos will make money regardless of the outcome of the game. The frequent player must win at approximately 52.4% in order to break even and in excess of that to make money over the long term.

A complete betting strategy will consist of two parts. First, the bettor must predict the outcome of an upcoming game. Second, he must use that information to decide if the game is safe to bet on, which team to bet on, and how large the wager should be. This project focuses solely on the first (and more difficult) part of the process.

I used the complete set of play-by-play data for the 2010-2011 NBA regular season. I split my data in half so that the first 623 games were used for training and I predicted on the last 607 games. More training data will give a more accurate estimation of MLEs, but will leave less data to test on as there are a fixed number of games. The choice of half and half was arbitrary but reasonable in that neither the training nor testing sets were too small to be viable. I did not try to predict playoff games even though I had data because in general they do not follow the same pattern as regular season games. Historical data on the Las Vegas betting lines was used as a point of comparison to evaluate how well my methods were able to predict spread.

Each individual game can possibly have a number of different outcomes with respect to the difference in score of the two participating teams. For example, the home team could win by four points, lose by twelve points, lose by seven points, etc. In general, teams cannot tie and no team will ever win by more than 100 points, so the possible number of outcomes is at most 200. We can think of a probability mass function existing on the set of possible outcomes for each game. If we somehow knew it exactly, we could calculate the exact chance that each team would beat the spread and we

would either bet on the favored team or not bet at all if the chance of winning was not high enough. Unfortunately we will never know the exact mass function, but we can try to estimate it empirically.

To do this, we will develop a way of independently sampling from a distribution that approximates the one that we are trying to estimate. We can then realize this approximating distribution by taking many samples. The method for taking a sample will be to simulate a single basketball game between the two desired teams.

When a team is in possession of the ball, there are only a few things they can do with it that will have either a direct or indirect impact on the score. Either they will attempt a shot, draw a shooting foul, or turn the ball over. If a shot is attempted, it either goes in, gets rebounded by the offense, or gets rebounded by the defense. If the team draws a shooting foul, two free throws are taken. If the second free throw is missed, the rebounding team will gain possession of the ball. If the offense turns it over, then by definition the defense has taken control of the ball. Eventually, the team that started with the ball will relinquish possession after possibly scoring some points. This constitutes one full possession. Teams alternate possession until time expires, and the final score is the summation of points accumulated during each possession. If we assume that each possession is an independent event, then the task of modeling an entire game boils down to modeling a single possession. We know that in practice, successive possessions are not necessarily independent. For example, teams may employ different strategies based on the time remaining and running score. This would mostly come into play at the end of a game. We make this simplifying assumption because modeling the dependencies of each possession is rather difficult and for most possessions, assuming independence is reasonable.

To model a single possession, we use the framework described above. Given the set of possible outcomes, we generate a possession by iterated random draws from different multinomial distributions. The parameters of each distribution, which are a set of probabilities that given events happen, are estimated from the data using maximum likelihood. For example, suppose that 10% of the Miami Heat's possessions end in a turnover and 20% result in a shooting foul. To model one of their possessions, we would draw first from a $U \sim \text{uniform}(0, 1)$ distribution. If $0 < U < .1$, then the result would be a turnover. If $.1 < U < .3$, there would be a shooting foul. We would then use different multinomial distributions to decide how many shots were made, and if the second shot was missed, who gets the rebound. The chances of these events occurring would be estimated from the Heat's past performance. If $U > .3$, then we know that the possession must end in either a made shot or a defensive

rebound. We draw from more distributions to decide how far away from the basket the shot is taken from, whether the shot goes in or not given the distance, and if it misses who gets the rebound. If it is determined that the offense rebounds, they will shoot again as we have decided by the very first draw that the possession did not end in a turnover or a foul. Finally, we can also estimate how long a possession lasts based on how long the Heat usually take for a single possession.

We must also consider the ability of the opposing team to defend. I calculate defense as modifiers to the chance that the offense scores given that they have taken a shot. First, I use the data to estimate the average field goal conversion rate for each team from each of seven distances. Then for each team and each game, I calculate how good the defense was versus the average. For example, suppose the Knicks convert long range three point shots at 30%. When the Heat played the Knicks, the Knicks converted long range three point shots at 35%. For this game, the Heat's defensive modifier for long range three point shots would be +5%. I then average across all games to get a set a seven defensive modifiers for each team. Whenever a team takes a shot, a defensive modifier is applied to that team's average conversion rate to come up with the true conversion rate.

This fully describes the methods I have employed. Figure One shows an empirical distribution generated by my model for a single game. One benefit of this type of model is it can be built upon and refined with ease. For example, we have assumed for simplicity that each team has an equivalent strength of schedule. If this were not the case, we would need to account for it. That is, it may be possible that the Heat and the Knicks convert short jump shots at 55% even though the Heat shoot short jump shots better than the Knicks. It just so happens that the Heat have played tougher defensive teams and therefore the two realized conversion rates are equal. We could try to estimate the strength of schedule of each team and then apply modifiers to conversion rates like we did for defense. While calculating the actual strength of schedule modifiers would be a project unto itself, applying them would be trivial.