AP Poll Ranking Learning Algorithm

Completed for Northwestern EECS 349, taught by Professor Doug Downey
Justin Savin | Nicholas David | Aaron Kamel
justinsavin2019@u.northwestern.edu | nicholasdavid2019@u.northwestern.edu | aaronkamel2019@u.northwestern.edu

College Basketball has quickly become a global pursuit. Every week, enthusiasts from around the world anticipate the release of the AP Poll, a ranking of 25 Division One Basketball teams, indicating the hierarchy in NCAA Division One Basketball. Often the best predictor in whether a team is to make the NCAA tournament, the AP Poll is an amalgamation of votes from hundreds of sports writers, and is thus a data-rich set. Our project centered around the question: Is there a pattern to this data-rich set; specifically, can we model whether a team will be ranked at the end of the season--cementing their place at the top of the college basketball hierarchy--based on common traits of teams that have finished the season ranked?

To begin, we gathered data from DataWorld (https://data.world/mkearney/ncaa-mens-cbb-teams) that lists common metrics for evaluating teams:

school--Name of the Division One college basketball program.

conf--Name of the Division One college basketball conference a program is a part of.

w--Number of games a Division One basketball team won in a specific year.

l--Number of games a Division One basketball team lost in a specific year.

wl--Win Loss Percentage, calculated by dividing the number of wins a Division One basketball team won in a specific year by the number of total games played.

srs--A ranking that takes into account average point differential and strength of schedule. The rating is denominated in points above/below average, where zero is average. Non-Division One games are excluding from the ratings.

sos--Strength of Schedule; a measure of how “difficult” a schedule is by denominating points over or under average, where zero is average. Non-Division One games are excluding from the ratings.

pts_for--Average points scored per game by a Division One basketball team.

pts_vs--Points scored against per game by a Division One basketball team.

pts_total--Summation of pts_for and pts_vs.

ap_pre--A number, 1-25 or 30 (not ranked) indicating whether a Division One basketball team was ranked to start the season and--if so--what a team’s rank was.

ap_high--A number, 1-25 or 30 (not ranked) indicating the highest rank on the AP Poll a Division One basketball team reached in a given season.

pts_diff--A difference between pts_for and pts_vs.

A multitude of steps were taken to best prepare this data for analysis. The first of which was to eliminate any extraneous variables—such as rk, ncaa_result, and coaches—not listed above so as to maintain accuracy, relevance, and functionality within Weka experiments. Additionally, to avoid oversimplification, we created a binary variable "ranked_final?" which displays a 1 if a Division One Basketball team is ranked following the final week of the regular season, and 0 if not. Inclusion of the ranked_final? attribute brings the total number of features to 14.

After modifying the data and features, we divided our data into two subsets: Testing Data and Training Data. The Training Data records all entries for the teams recorded up through the 2013-2014 season, and consists of 5,827 entries. Similarly, the Testing Data records all entries in during the 2014-2015 and 2015-2016 seasons, and consists of 620 entries.

To determine the best classifier for the data, we trained a variety of learners over the Training Data, evaluating their performance using 10-Fold cross validation. We first tested our data against ZeroR to serve as a baseline against which we would compare our final learner. Next, we tried an IBK learner thinking that teams with similar schedules, conferences, and win-loss records would finish the season ranked in a similar position. Cognizant of the fact that certain attributes were dependent on one-another, we tested our data with a Naïve Bayes learner out of curiosity as to whether the independence assumption would significantly improve accuracy. Finally, we moved to a series of Decision Tree learners, the motivation being that a decision tree best represented our hypothesized evaluation method; our hypothesis was correct, a J48 Decision Tree yielded the highest accuracy when trained over our data. The results are summarized below:

Learning Method	Accuracy	Precision	Recall
ZeroR	91.8325%	.000—Ranked Teams .918—Unranked Teams	0—Ranked Teams 1—Unranked Teams
IBK	96.3979%	.745 .983	.795 .978
Naïve Bayes	96.741%	.698 1.0	1.0 .965
Random Forest	98.164%	.918 .987	.851 .993
Random Tree	97.5986%	.841 .987	.841 .987
J48 Decision Tree	98.97%	.932 .994	.932 .994

As shown above, a J48 Decision Tree has the greatest accuracy, precision, and recall when trained over the data and evaluated via 10-Fold cross validation. We therefore chose to adopt a J48 Decision Tree with splits at the features “ap_high” and “l”; a graphic depiction of our tree, as well as quantitative results when applied to the Testing Data, are shown below:

Learning Method	Accuracy	Precision	Recall
J48 Decision Tree—Testing	97.42%	.8863—Ranked .9809—Not Ranked	.78—Ranked .9913—Not Ranked

While analyzing results, we were initially startled by the simplicity of the tree without sacrificing accuracy; in particular, the inclusion of solely two attributes was cause for concern. It is possible that the inclusion of only two attributes indicates that Weka was at a local minimum. To further investigate this situation, we again trained and tested our data on a J48 Decision Tree, this time removing the ap_high attribute; the motivation for this experiment was to see if—in absence of the assumption that the season had concluded—an increasingly complicated tree consisting of a greater number of attributes was formed. Our hypothesis was confirmed by Weka’s J48 Tree Output:

```
srs <= 12.7
|   w <= 22: No (4671.0/4.0)
|   w > 22
|   |   wl <= 0.846
|   |   |   srs <= 9.63: No (324.0/2.0)
|   |   |   srs > 9.63
|   |   |   |   l <= 10
|   |   |   |   |   sos <= 6.8
|   |   |   |   |   |   l <= 7
|   |   |   |   |   |   |   sos <= 3.77
|   |   |   |   |   |   |   |   srs <= 10.18
|   |   |   |   |   |   |   |   |   sos <= 0.81: No (2.0)
|   |   |   |   |   |   |   |   |   sos > 0.81: yes (4.0)
|   |   |   |   |   |   |   |   srs > 10.18: No (17.0/1.0)
|   |   |   |   |   |   |   sos > 3.77: yes (5.0)
|   |   |   |   |   |   l > 7: No (55.0/9.0)
|   |   |   |   |   sos > 6.8: yes (6.0)
```

```
|  |  |  |  l > 10: No (38.0/3.0)
|  |  wl > 0.846
|  |  |  wl <= 0.871
|  |  |  |  srs <= 8.85: No (7.0)
|  |  |  |  srs > 8.85
|  |  |  |  |  l <= 4: yes (3.0/1.0)
|  |  |  |  |  l > 4
|  |  |  |  |  |  pts_for <= 73.4: No (2.0)
|  |  |  |  |  |  pts_for > 73.4: yes (5.0)
|  |  |  wl > 0.871: yes (9.0/1.0)
srs > 12.7
|  l <= 10
|  |  srs <= 16.89
|  |  |  l <= 8
|  |  |  |  pts_diff <= 10.4: yes (55.0/3.0)
|  |  |  |  pts_diff > 10.4
|  |  |  |  |  srs <= 13.89: No (7.0/1.0)
|  |  |  |  |  srs > 13.89
|  |  |  |  |  |  l <= 6: yes (12.0)
|  |  |  |  |  |  l > 6
|  |  |  |  |  |  |  pts_total <= 141.7
|  |  |  |  |  |  |  |  pts_vs <= 63
|  |  |  |  |  |  |  |  |  w <= 27: No (2.0)
|  |  |  |  |  |  |  |  |  w > 27: yes (3.0/1.0)
|  |  |  |  |  |  |  |  pts_vs > 63: yes (5.0)
|  |  |  |  |  |  |  pts_total > 141.7: No (4.0)
|  |  |  l > 8
|  |  |  |  sos <= 5.57: No (22.0/2.0)
|  |  |  |  sos > 5.57
|  |  |  |  |  wl <= 0.692
|  |  |  |  |  |  pts_vs <= 67.7: No (9.0/1.0)
|  |  |  |  |  |  pts_vs > 67.7
|  |  |  |  |  |  |  pts_for <= 79.3: yes (5.0)
|  |  |  |  |  |  |  pts_for > 79.3: No (3.0/1.0)
|  |  |  |  |  wl > 0.692: yes (67.0/17.0)
|  |  srs > 16.89: yes (271.0/6.0)
|  l > 10
|  |  srs <= 16.94: No (183.0/16.0)
|  |  srs > 16.94
|  |  |  pts_vs <= 70.5: No (25.0/8.0)
|  |  |  pts_vs > 70.5: yes (7.0)

Number of Leaves  :   30

Size of the tree :    59
```

The discrepancy in tree complexity indicates we are not stuck at a local minimum, and corroborates our suspicions of a difference in the evaluation methods taken by AP Poll voters with respect to whether the season has been completed or not. In short, depending on how much of the season has been completed, AP Poll Voters rank teams differently, as evidenced by the vastly different J48 Decision Trees output by Weka. In the future, an interesting experiment would be to evaluate J48 Decision trees across various weeks in a NCAA Division One College Basketball season, and evaluate which features have the highest information gain. In this way, we can predict more accurately across various points in a season whether an NCAA Division One College Basketball team will be ranked by the AP Poll or not.

Nicholas David—Dataset manipulation and classification

Aaron Kamel—Host and Designer of final project website

Jusitn Savin—Author of final project report