Possible Models to use
ANN (FFNN)
- apply a log sigmoid transfer function as the activation function
- Relatively slow training
- one hidden layer

LSTM
- Type of Recurrent Neural Network to look at a sequence of games in order to predict the scores, rather than making a prediction based on a single game (FFNN)
- assume that a player's PER scores in successive games are co-dependent as a player
- might be in certain shape and therefore achieve similar results in those games
- Helpful for evaluating series of games

Naive Bayes
- probabilistic classifier, which means it predicts on the basis of the probability of an object
- Has an independence assumption

LMT: Logistic Model Tree
- combines logistic regression model and a decision tree structure to derive a single tree for classification
- The combination of two complementary algorithms produces an accurate and interpretable classifier by combining the advantages of both logistic regression and tree induction. The issue arises because as model accuracy increases so does model complexity, at the cost of interpretability.
- LMT aims for both high predictive accuracy and interpretability by combining the two algorithms in a single tree.
- LMT uses a boosting approach called LogitBoost to incrementally refine the logistic regression models along their corresponding paths from the root to the leaves. The additive modeling of logistic regression by LogitBoost provides a natural way of `incremental learning' of the leaf models.
- Thus, the final logistic regression models at the leaf nodes can reflect both the global influences on the tree structure and the specific local effect of the partitioned data space at each node.

Naives vs LMT vs ANNs
- Naive has independence assumption
- Naive bayes is a generative model whereas LR is a discriminative model. Naive bayes works well with small datasets, whereas LR+regularization can achieve similar performance. LR performs better than naive bayes upon collinearity, as naive bayes expects all features to be independent
- In general, logistic regression models are less prone to overfitting than are ANNs because they involve simpler relationships between the outcome variable and predictor variables

Ten fold cross validation

- Having a lower K means less variance and thus, more bias, while having a higher K means more variance and thus, and lower bias.
- Higher computational time with 10 vs 5

Feature Selection
Multiple Regression

- Five main assumptions underlying multiple regression models must be satisfied: (1) linearity, (2) homoscedasticity, (3) independence of errors, (4) normality, and (5) independence of independent variables

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.053092   0.199053  -0.267 0.789813
Home         0.112837   0.040253   2.803 0.005293 **
TP           0.078227   0.010511   7.443 5.57e-13 ***
TPA         -0.038323   0.004459  -8.594  < 2e-16 ***
ORB          0.002251   0.004577   0.492 0.623201
DRB          0.039215   0.004300   9.119  < 2e-16 ***
STL          0.026940   0.006892   3.909 0.000108 ***
TOV         -0.026717   0.005203  -5.135 4.31e-07 ***
PF          -0.015686   0.004481  -3.501 0.000513 ***
---
```

Correlation Feature Set

- The logic behind using correlation for feature selection is that good variables correlate highly with the target. Furthermore, variables should be correlated with the target but uncorrelated among themselves
- It evaluates the worth of an attribute by measuring the correlation (Pearson's) between it and the class. Nominal attributes are considered on a value by value basis by treating each value as an indicator. An overall correlation for a nominal attribute is arrived at via a weighted average.

Ripper Algorithm

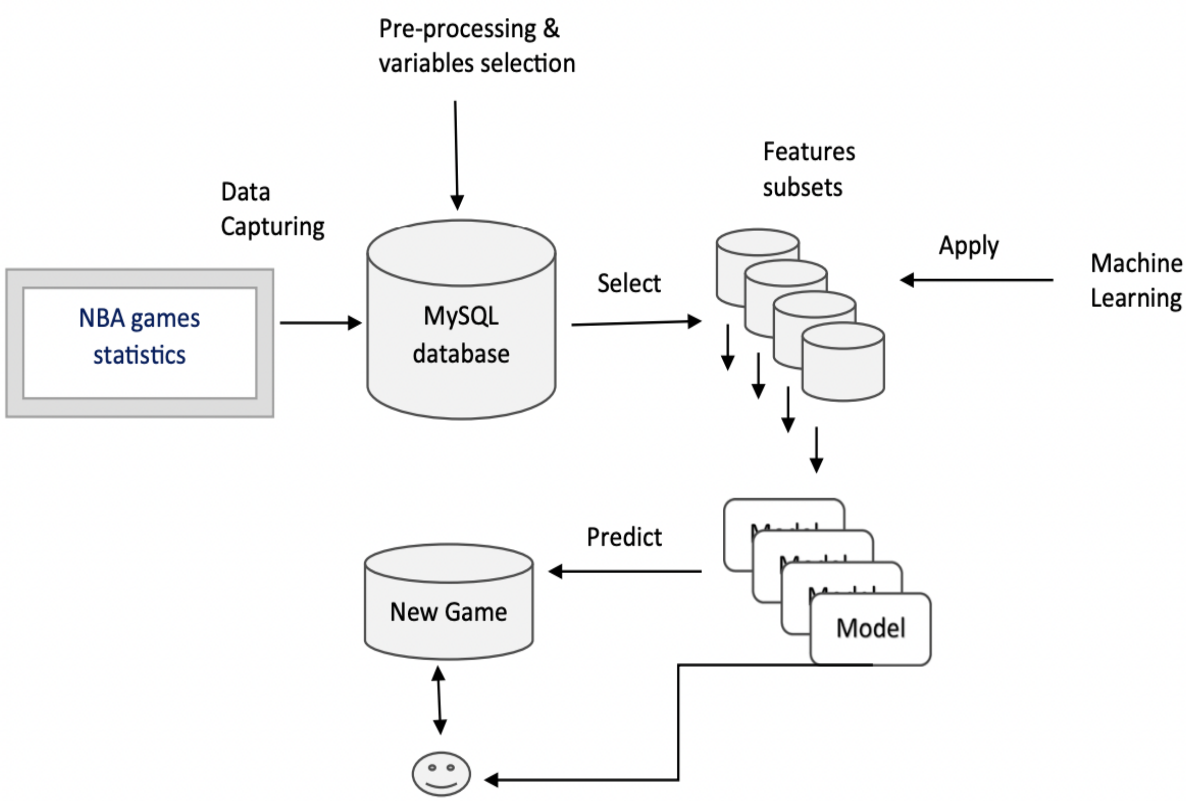
- The RIPPER (repeated incremental pruning to produce error reduction)
- Grow, prune, optimize
- The first step uses a ‘separate and conquer’ method to add conditions to a rule until it perfectly classifies as a subset of data. Just like decision trees, the information gain or “entropy” criterion is used to identify the next splitting attribute. When increasing a rule’s specificity no longer reduces entropy, the rule is immediately pruned. Until reaching stopping criterion step one and two are repeated at which point the whole set of rules is optimized using a variety of heuristics

- BENEFITS: It works well with noisy datasets as it uses a validation set to prevent model overfitting. The new rule is pruned based on its performance on the validation set. It also works well with imbalanced class distributions

NBA game result prediction using feature analysis and machine learning
Fadi Thabtah, Li Zhang & Neda Abdelhamid, Annals of Data Science (2019)

Objectives

- Detect the influential features set that impacts the outcome of games in the NBA (Filters: Multiple Regression, Correlation Feature Set; Rule Induction Algorithm: RIPPER repeated incremental pruning to produce error reduction)
- Once the features set is known, utilize that information to build a prediction model (ANN, Naive Bayes, Logistic Model Tree)
- proposes a new intelligent machine learning framework for predicting the results of games played at the NBA by aiming to discover the influential features set that affects the outcomes of NBA games



Dataset: NBA Finals Team Stats

- Data from NBA finals, including game-by-game team totals from 1980 - 2017
- 22 variables, 430 instances, 11 teams
- Training dataset includes details of independent variables (matches played in home

ground, percentage of field goals made, etc)
- Limitations: smaller variance, upset rate of ~20%

Preprocessing

classification task → format of outcome "WIN" (1=WIN, 0=LOSS) from numeric to nominal (WEKA filter tool of NumtoNom)

continuous values are discretized (WEKA discretization filter)

Feature Selection

- Identify set of influential features (variables) by removing irrelevant variables → reduces input dataset dimensionality → improve learning process and model's performance (predictive accuracy)
- 3 methods: Filters: Multiple Regression, Correlation Feature Set; Rule Induction Algorithm: RIPPER
- 7-8 features selected, 14-15 ignored → dataset reduced by more than ½
- DRB: feature selected in all methods

Prediction Models
- 3 machine learning algorithms selected: ANN, Naive Bayes, Logistic Model Tree
- **ANN:** FFNN, utilize multiple independent variables and their linked weights to build a network structure, repeating training by adjusting the weights until the desired output is reached
- **Naive Bayes:** probabilistic classifier that employs joint probabilities computed from the labelled observations to forecast the class of a test data
- **LMT:** combines logistic regression model and a decision tree structure to derive a single tree for classification

Experimental Analysis

- All models were tested based on ten-fold Cross Validation
- Evaluation metrics used to assess model performance: predictive accuracy, recall, precision, and harmonic mean (F1)

Accuracy: measure of how effective the model is at predicting outcomes; Accuracy = (TP + TN)/(TP + FP + FN + TN)

Precision: measure for positive prediction; Precision = TP/(TP + FP)

Recall: measure of correctly-predicted positive observations to all observations in the positive class; Recall = TP/(TP + FN)

F1 Score: weighted average of Precision and Recall; F1 Score = 2 ∗ (Recall ∗ Precision)/(Recall + Precision)

Results

- Dataset B: lowest performance, multiple regression is not a suitable method for feature selection for this
- **Dataset D (feature selection through the Ripper algorithm) achieved best classification performance across the 3 different machine learning techniques**
- Accuracy rates obtained using the three different models (Naïve Bayes, ANN, and LMT) built on features set D revealed higher values **(80%, 80%, and 83%)** compared to the other features sets
- F1, precision, and recall best rates were obtained by LMT algorithm from dataset D **(83%, 83%, and 83%)** respectively.

Improve results through feature selection (2-4% increase in prediction accuracy as opposed to whole Dataset A) → prioritize certain strategies to improve team's playing capabilities

Conclusion + Future Work

This paper developed a framework based on machine learning and feature selection to deal with the problem of result prediction of NBA games. After investigating various machine learning techniques to build prediction models using different features sets obtained by feature selection methods, we arrive at a conclusion of significant factors affecting the outcome of NBA matches

- DRB feature (defensive rebounds), TPP (three-point percentage), FT (free throws made), FGP (field goal percentage), and TRB (total rebounds)

Other Applications & Future Work

- Game predictions vs. Daily Fantasy betting (constraint problem in terms of risk, last x games)
- Other influential factors for feature selection, weighting individual players performance (individual player statistics / Player Efficiency Ratings for minutes played), coach performance, injuries
- deep learners for instant model adjustment while the game is playing live