Analysis of Poker Hands

Group Members:
Brian Sebastiao - 214 182 422
Albert Hakobyan - 213 775 630
We Lan Wu - 212 350 161
Andy Tran - 213 797 832

Table of Contents

1 Introduction

Poker is a game of luck and skill. It combines the nature of luck by randomly dealing a 52 card deck among 2-9 players. The game usually involves some form of gambling, where money is won if a player wins. The problem to be addressed is assuming a 5-card stud game, we want to maximize our chances of winning based on the cards we currently have in hand. Using this knowledge, a player can leverage their decision making to minimize their losses when making a bet. To do this, we need to apply different algorithms to a dataset of poker hands.

There are different "hands" in poker. These hands are combinations of cards that hold some value to win the game. This value is what determines whether you win the round or lose to another player's higher valued hand. The different hands are listed (from highest to lowest) as followed:

- Royal Flush: The highest possible value in Poker, consisting of the highest cards in the game, all with the same suit. The cards are 10, Jack, Queen, King, Ace
- Straight Flush: Second highest possible hand value. This consists of 5 cards, all of incrementing value with the same suit. These cards can be 5,6,7,8,9.
- Four of a Kind: A hand consisting of 4 of the same card. An example is 4 Queens.
- Full House: A hand consisting of 3 of the same card, with a pair of the same card. An example is: 3 10s and 2 4s.
- Flush: Cards consisting of the same suit, but not in any sequence. An example is 5,9,2,Ace,Queen
- Straight: This consists of 5 cards in a sequence, however they are not of the same suit. An example is 5,6,7,8,9.
- Three of a Kind: Three cards, all of the same number or rank. An example is 3 7s, a King and a 3.

- Two Pair: A hand consisting of two different pairs, an example is: 2 5s, 2 4s, and a King.
- One Pair: Two cards of the same rank. An example is 2 Aces, 6,2, King
- High Card: When you do not have any of the hands above, the highest card in your hand is what you compare to other players. For example 3, Queen, Jack, 2, 5. In this case, the Queen would be compared to all other player's hands. In regards to our data problem, high cards are not considered a poker hand.

Based on these hands, we can agree that the best possible hand we can achieve is the royal flush. This however, is extremely unlikely to occur as it requires a very specific set of cards to accomplish.

2 Problem Definition

2.1 Data Set

The dataset the group decide to use can be found at this link: (https://archive.ics.uci.edu/ml/datasets/Poker+Hand)

The following are different characteristics of the dataset:

Number of Instances: 1025010 (1000000 - test set and 25010 training set)

Number of Attributes: 11

Class Attribute: 0-9 and is representative of ranking of the poker hand.

S = Suit of Card: This represents the suit of the card labelled 1-4

1 = Heart

2 = Space

3 = Diamond

4 = Clubs

C = Rank of Card: Represents the rank of the card labelled 1-13.

1 = Ace

2 = Two

3 = Three

…

11 = Jack

12 = Queen

13 = King

Class = Poker Hand

0: Nothing in hand; not a recognized poker hand

1 = One pair; one pair of equal ranks within five cards

2 = Two pairs; two pairs of equal ranks within five cards

3 = Three of a kind; three equal ranks within five cards

4 = Straight; five cards, sequentially ranked with no gaps

5 = Flush; five cards with the same suit

6 = Full house; pair + different rank three of a kind

7 = Four of a kind; four equal ranks within five card

8 = Straight flush; straight + flush

9 = Royal flush; {Ace, King, Queen, Jack, Ten} + flush

Figure 1.1: Histogram of All Hands (Training Set)

Figure 1.1 shows from 0-9, the frequencies are 12493, 10599, 1206, 513, 93, 54, 36, 6, 5, and 5. The test set contains a total of 25010 frequencies. It can be noted that as the card rank gets higher, the probability of getting said hand is exponentially lower; with ranks 4-9 being invisible in the histogram.

Figure 1.2: Histogram of Winning Hands Only (Training Set)

Figure 1.2 is similar to Figure 1.1 with only winning hands being displayed.

2.2 Data Preprocessing

In regards to the training data, we cleaned it to be more easily read/managed - raw data was meant to simulate a hand of 5 cards, one suit and one rank. However, plaintext showed it as a string of texts consisting of 11 numbers delimited by a comma from 1-4, 1-13, and 0-9 indicating in order: the suit, the card rank, and the card combination class found. Figure 2.1 shows the first two rows in the training data.

S1	C1	S2	C2	S3	C3	S4	C4	S5	C5	Class
1	10	1	11	1	13	1	12	1	1	9

2	11	2	13	2	10	2	12	2	1	9

Figure 2.1: Plain data shown in table

As a result, the string had to be split by the delimited comma to create ten different columns. For S1-S5, each of the corresponding numbers (1-4) was replaced by the first letter in the suit name (1 - **H**eart, 2 - **S**pade, 3 - **D**iamond, 4 - **C**lubs). For C1-C5, we did something similar to indicate face cards and ace (1 - Ace, 11 - Jack, 12 - Queen, 13 - King). Finally, the last column was replaced with the true names of the hand ranking (0 = None, 1 = Pair, … 8 = Straight Flush, 9 = Royal). Once completed, we then concatenated the values of each suit and rank to form a single card. This reduces the number of attributes from 11 to 6. Figure 2.2 shows the revised data from Figure 2.1.

Card 1	Card 2	Card 3	Card 4	Card 5	Hand
H10	HJack	HKing	HQueen	HAce	Royal
SJack	SKing	S10	SQueen	SAce	Royal

Figure 2.2: Cleaned training data

By cleaning the data, it makes the data easier to interpret and analyze. The first hand in Figure 1.2 can be interpreted as Card 1 = Ten of Hearts, Card 2 = Jack of Hearts, Card 3 = King of Hearts, Card 4 = Queen of Hearts, Card 5 = Ace of Hearts, and the Hand = Royal).

2.3 Issues and Difficulties

When attempting to apply the apriori algorithm, it was discovered that the dataset of a deck of cards with 13 values and 4 suits have no relation to each other and the probability of one card appearing with another would not increase the likelihood of another showing up in pairs or higher. The data set chosen was also found to be a very

challenging dataset for classification algorithms, with neural networks being one of few choices. In regards to clustering algorithms, we also had the issue of having high-dimensionality data. This made clustering difficult as there were 52 different attributes.

3 Algorithms

3.1 Algorithm 1 - Apriori

```
> rules = apriori(values, parameter = list(supp = 0.001, conf = 1))
Apriori

Parameter specification:
 confidence minval smax arem  aval originalSupport maxtime support minlen
          1    0.1    1 none FALSE            TRUE       5   0.001      1
 maxlen target   ext
     10  rules FALSE

Algorithmic control:
 filter tree heap memopt load sort verbose
    0.1 TRUE TRUE  FALSE TRUE    2    TRUE

Absolute minimum support count: 25

set item appearances ...[0 item(s)] done [0.00s].
set transactions ...[15 item(s), 25010 transaction(s)] done [0.00s].
sorting and recoding items ... [15 item(s)] done [0.00s].
creating transaction tree ... done [0.00s].
checking subsets of size 1 2 3 4 5 done [0.00s].
writing ... [0 rule(s)] done [0.00s].
creating S4 object  ... done [0.02s].
Warning message:
Column(s) 1, 2, 3, 4, 5 not logical or factor. Applying default discretization (see '? discretizeDF').
```

Figure 3.1: Apriori Code

The program used is R, and initially apriori was used to determine if there would be a relation between a card appearing when there is a winning combination of pair, three of a kind, flush, or full house. This was done as the hypothesis is with a five-card hand, a poker player would likely discard the lower card values in order to maximize their chances of winning. As a result, using apriori, it was expected that in hands with duplicate card values, the probability of also containing a low card should be higher as they are more likely to be drawn. This was modelled by having a minimum value of 0 to allow for any card value to appear while requiring a confidence of 1. The confidence 1 is mandatory as pairs and above all require at least one card with the same value to appear in a 5 card hand. However, as it can be seen with apriori, no rules were generated as every card in the deck had an equal chance of getting a winning combination. It seemed

that despite cards with lower values are seen as bar for kickers, the likelihood for the low card to be discarded or drawn is not changed as the low cards have the same probability of being a part of a pair at the start as any other card value. After discarding cards, the probability of drawing any other card is the also the same as the probability of drawing a card with low value. Thus no rules were able to be generated despite the initial hypothesis. With these results, it can be concluded that it does not matter what cards are kept after the initial hand deal, as it has no effect on what cards would potentially be drawn, instead it should be look to maximize what strength the original hand currently has (pair, high card, open end straights).

3.2 Algorithm 2 - Density-Based Clustering

```
>  prob = read.csv("prob.csv", fileEncoding="UTF-8-BOM")
> p = ggplot(prob,aes(x=value,y=hand)) + ggtitle("Plot of Value vs Hand") + xlab("Value") + ylab("hand") + ylim(0,10) + xlim(0,13)
> p1 = p +
+   geom_point(alpha = 0.01, colour="orange") +
+   geom_density2d() +
+   theme_bw()
> p2 = p +
+   stat_bin_hex(colour="white", na.rm=TRUE) +
+   scale_fill_gradientn(colours=c("purple","green"),
+                        name = "Frequency",
+                        na.value=NA)
> grid.arrange(p1,p2,ncol=2)
>
```

Figure 3.2: Density-Based Clustering Code

In this scenario there was an attempt to visualize the distribution of a specific card compared to the frequency of the type of hand it could potentially make. It is hypothesized that the more extreme values are less likely to form a straight hand as they cannot form an open-ended straight as easily, while royal flush is exclusive only to the higher ranked cards. Visualized it was performed with colors, with higher density being green, with a gradual transition to purple for rarer distributions. The y-axis represents the strength of the hand, with 0 being no combination and 9 being the highest (royal flush). The x-axis represents the value of the card that is represented, with the higher face cards(Jack, Queen, King) being represented by 11-13. Perhaps better visualized by the color plot, as the density-wave plot is difficult to see due to the extremely large amounts of “none” and “pair” hands. We observe that as expected, royal flush is only present for face cards and ace, while flushes are more common for 5 and 8; both of which are more

likely to be opened up for an open ended straight. What was surprising however, was that within the test of 25000 hands, card number 3, 6, 7, and 10 were more likely to have a pair than others. This perhaps could be due to 3, 6, 7, and 10 having a high likelihood to be kept if they are together for the hopes of an open ended straight. However, the more likely scenario would be resulting in having a pair instead due to lower probabilities of acquiring a straight or higher. An example would be a hand may consist of the cards 3, 6, and 7, so seeing the potential for a straight, the player keeps the cards. However, due to the deceptively low probability of drawing the exact two cards needed to complete a straight, the strength of the hand drops to either 1 or lower dramatically as opposed to just drawing a new hand of 5 cards. This seems to indicate that, at least for small samples, it is better to just aim for small hands to maximize chances of winning due to the extremely large chance of getting a pair or lower, as opposed to aiming for a high powered hand with a very low probability of success. It should be noted that if three cards are saved, the probability of drawing three of a kind would be extremely low, four of a kind would be impossible, and full house impossible. Looking at the graphs we can see that there exist no quadruples for the numbers between 3 to 8, as well as 9 to Jack since these cards are near the middle values and thus hold no possibility for a four of a kind should they be held in hopes of a straight. It should be noted however, that the highest value card A has matched four-of-a-kind as it seems to be possible that ace is held in order to win with high-card against nothing.

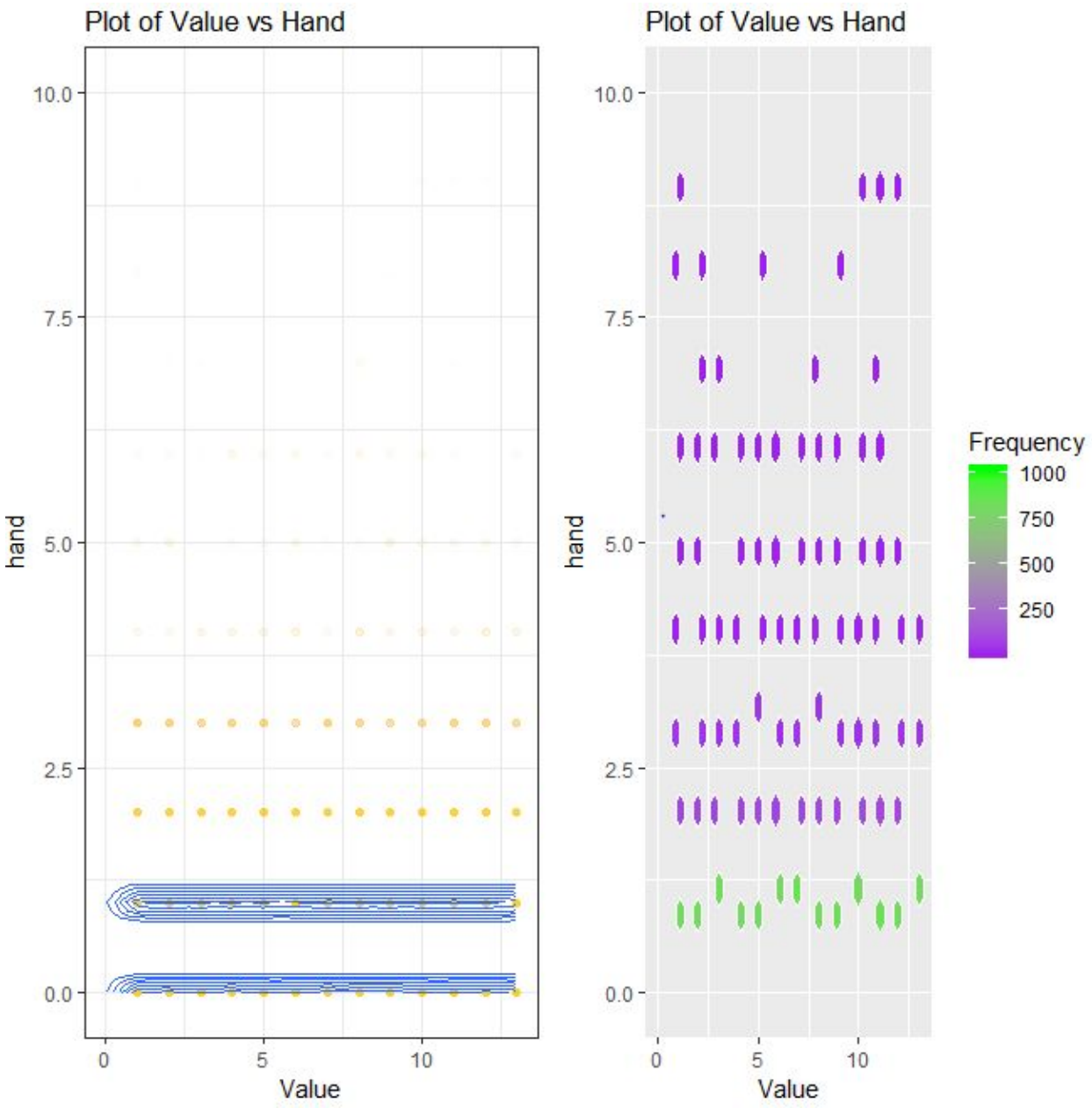

Figure 3.3: Clustering Graph

```
> colorsp = c("#313695", "#4575B4", "#74ADD1", "#ABD9E9", "#E0F3F8", "#FFFFBF","#FEE090", "#FDAE61", "#F46D43", "#D73027",
>  #smoothgraph
>  colorgrad=colorRampPalette(c(colorsp))
>  hand = prob$hand
>  value = prob$value
>  smoothScatter(x=value,y=hand, colramp=colorgrad, main="Plot of Value vs Hand", xlab="Value", ylab="Hand Strength")
```

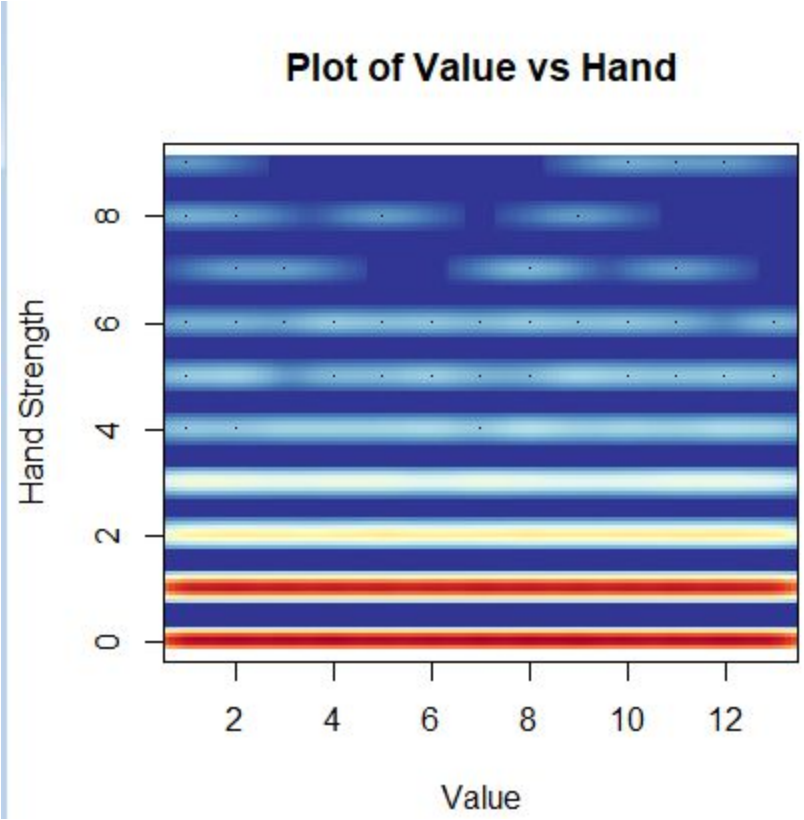

Figure 3.4: Heat-Based Graph

4 Training and Test Set Results

Hand	Frequency	Probability
Royal Flush	5	0.0200%
Straight Flush	5	0.0200%
Four of a Kind	6	0.0240%
Full House	36	0.1439%

Flush	54	0.2159%
Straight	93	0.3719%
Three of a Kind	513	2.0512%
Two Pairs	1206	4.8221%
One Pair	10599	42.3790%
Nothing	12493	49.9520%

Figure 4.1: Training Set Frequencies and Probability

Hand	Frequencies	Probability
Royal Flush	3	0.0003%
Straight Flush	12	0.0012%
Four of a Kind	230	0.0230%
Full House	1424	0.1424%
Flush	1996	0.1996%
Straight	3885	0.3885%
Three of a Kind	21121	2.1121%
Two Pairs	47622	4.7622%
One Pair	422498	42.2498%
Nothing	501209	50.1209%

Figure 4.2: Testing Set Frequencies and Probability

Hand	Possible Combinations	Probability
Royal Flush	4	0.00015%
Straight Flush	40	0.0015%
Four of a Kind	624	0.0240%

Full House	3744	0.1441%
Flush	5108	0.1965%
Straight	10200	0.3925%
Three of a Kind	54912	2.1128%
Two Pairs	123552	4.7539%
One Pair	1098240	42.2569%
Nothing	1302540	50.1177%

Figure 4.3: Possible Hand Combinations and Probability

Training Set	Test Set	Actual Combination
0.0200%	0.0003%	0.00015%
0.0200%	0.0012%	0.0015%
0.0240%	0.0230%	0.0240%
0.1439%	0.1424%	0.1441%
0.2159%	0.1996%	0.1965%
0.3719%	0.3885%	0.3925%
2.0512%	2.1121%	2.1128%
4.8221%	4.7622%	4.7539%
42.3790%	42.2498%	42.2569%
49.9520%	50.1209%	50.1177%

Figure 4.4: Comparison of Different Data

5 Evaluation and Comparison

In general mining for frequent patterns in our dataset proved to be difficult. With over 2.5 million possible poker hands, even with our test set of 1 million instances, not

all possible hand combinations can be taken into account for. This resulted in us being unable to find any patterns or frequent hands as there are just too many possible combinations.

In regards to clustering, due to the nature of our data set being a random sample of cards in a poker hand, we found it difficult to implement other partitioning methods as they cluster based on distance between objects. With 52 cards being in a standard deck, there are 52 outcomes that we had to deal with in our clusters. The high dimensionality of our data set made clustering difficult. We found that finding any meaningful cluster in a deck of card is pointless as the chance of receiving a certain card is random. Instead we decided to cluster based on rank of hand strength (0-9) and the rank of card (1-13). We looked at what rank of cards appear frequently in each rank of hand strength. By knowing this, we were able to make assumptions on what moves the player should do based on the rank of cards they have.

6 Work Division

Task	**People**
Data Processing	Andy Tran, Wei-Lan Wu
Algorithm 1 - Apriori	Wei-Lan Wu
Algorithm 2 - Density Clustering	Wei-Lan Wu
Modelling	Wei-Lan Wu
Probability Calculations	Andy Tran
Evaluating Algorithms	Andy Tran
Presentation	Brian Sebastiao, Wei-Lan Wu
Writing Report	Brian Sebastiao, Andy Tran, Wei-Lan Wu