Jane Street Market Dataset

Areeb Abubaker & Jordan Bickelhaupt

Presentation Outline

1. Research Problem
2. Cleaning and Exploration
3. Methodologies:
    a. Linear Regression
    b. Principal Component Analysis
    c. Multidimensional Scaling
    d. Cluster Analysis
    e. Cluster Modeling
4. Limitations
5. Conclusion

Research and Problem

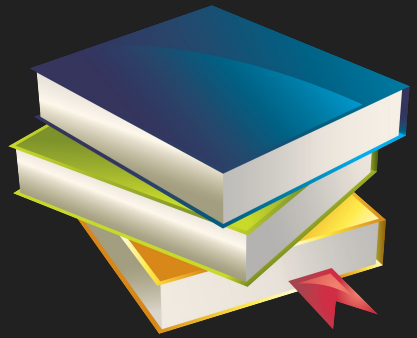
Question:

Can we determine what a good trade would look like?

Literature Review

Nguyen, T., & Huynh, N. (2019). Clustering Stock Market Data Using K-Means Clustering Algorithm. Retrieved from https://athena.ecs.csus.edu/~nguyethi/TN_NH_CSC177_Report.pdf used principal component analysis to identify kmeans clustering model techniques.

Sun, C. (2017, April 25). Application of K-Means Clustering and NeuralNetwork to Stock Return Prediction. Retrieved 2021, from https://cpb-us-w2.wpmucdn.com/blogs.baylor.edu/dist/d/4574/files/2018/01/project_presentation-1kol1u6.pdf utilized kmeans clustering with neural networks to generate predictions about stock performance.



About the Data

Kaggle Dataset source:

https://www.kaggle.com/c/jane-street-market-prediction

Dataset is composed of:
Time Series Response{0...4},
Date & ts_id time series ordering,
Features {0...129},
Response,
Weight

New variable created:
Response * Weight ---> R*W

~5 GB Dataset

Gave limitations in performance

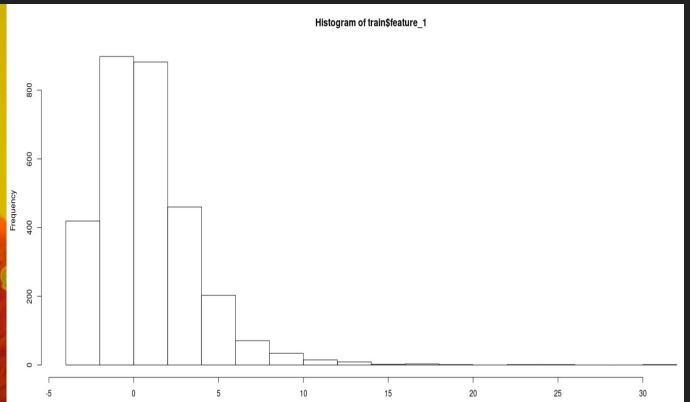
particularly with Multidimensional Scaling

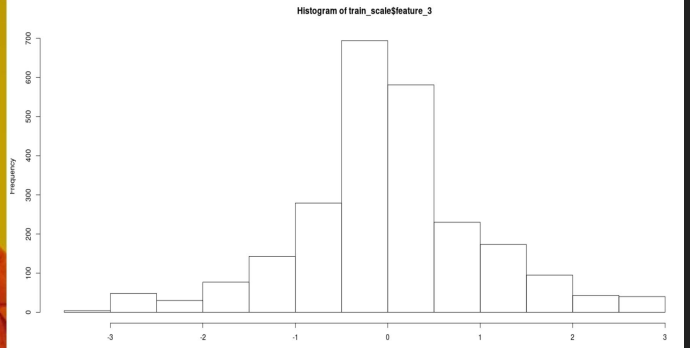
Solution: nrows = 3000

Data Cleaning

- Date and ts_id removed (impractical)
- resp{1...4} removed due to collinearity

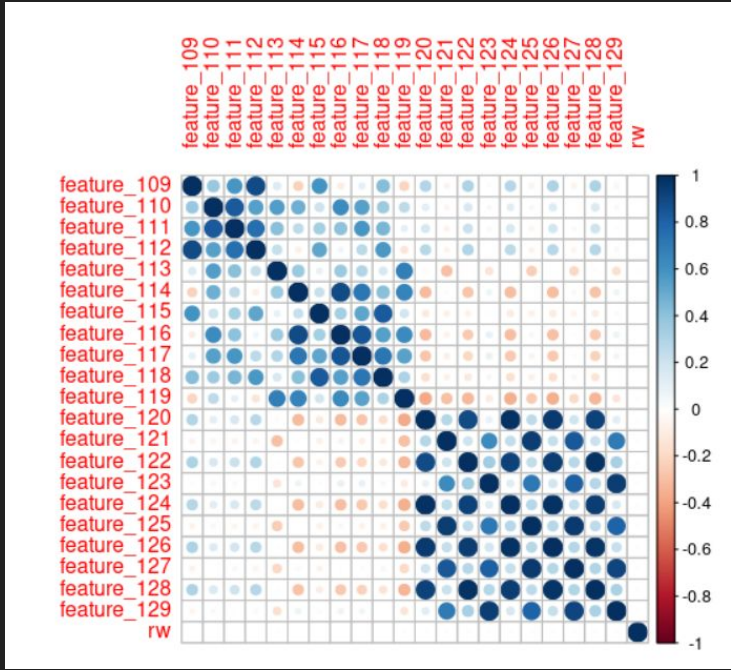
Dataset was scaled and
NA entries were removed.

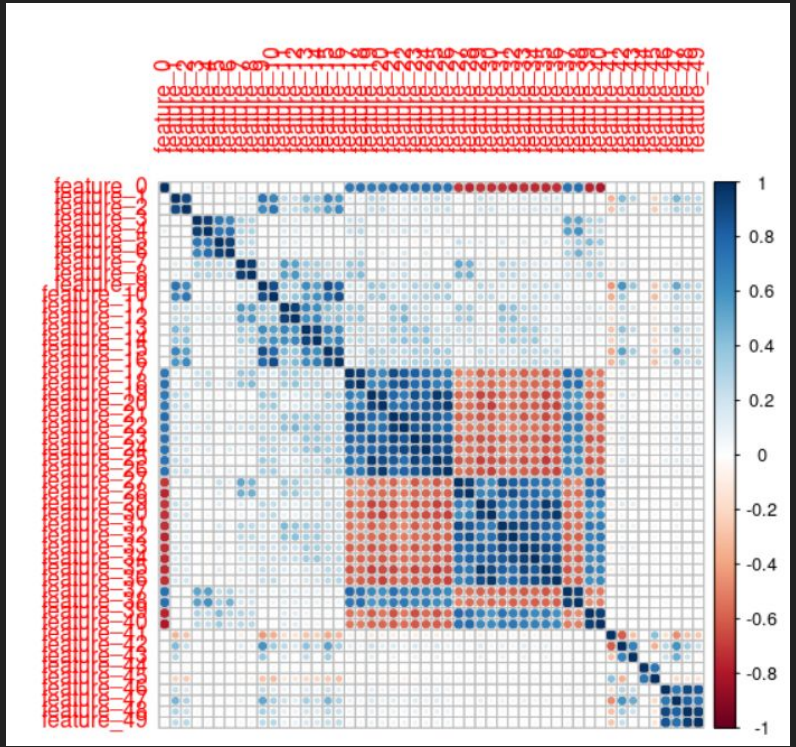




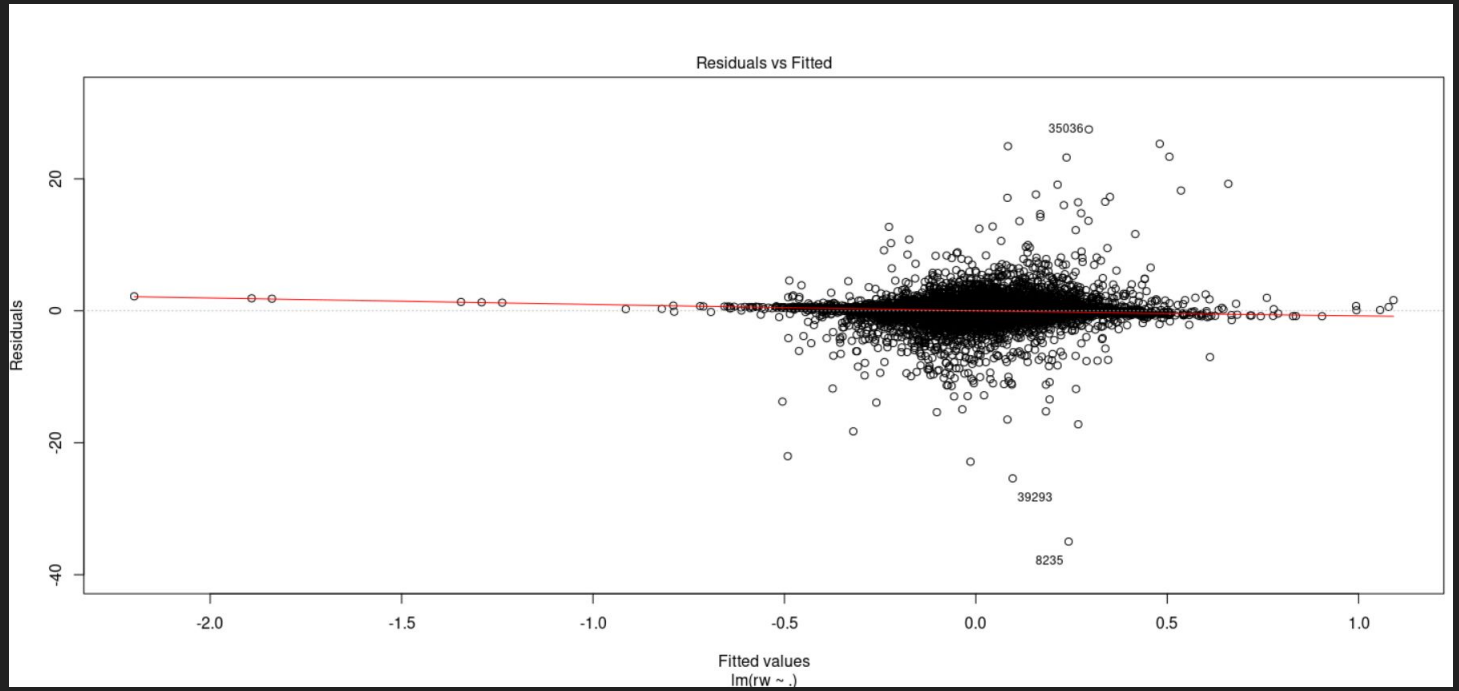
Data Exploration

Corrplots of first 50 features and last 21 features (including response * weight (RW))





Train(scaled) linear model ---> plot(ts_lm)



Linear Regression Validation

Linear Model:

lm(formula = rw ~ ., data = train_scale)

```
Residual standard error: 0.9732 on 2306 degrees of freedom
Multiple R-squared:  0.1035,    Adjusted R-squared:  0.05296
F-statistic: 2.048 on 130 and 2306 DF,  p-value: 1.406e-10
```

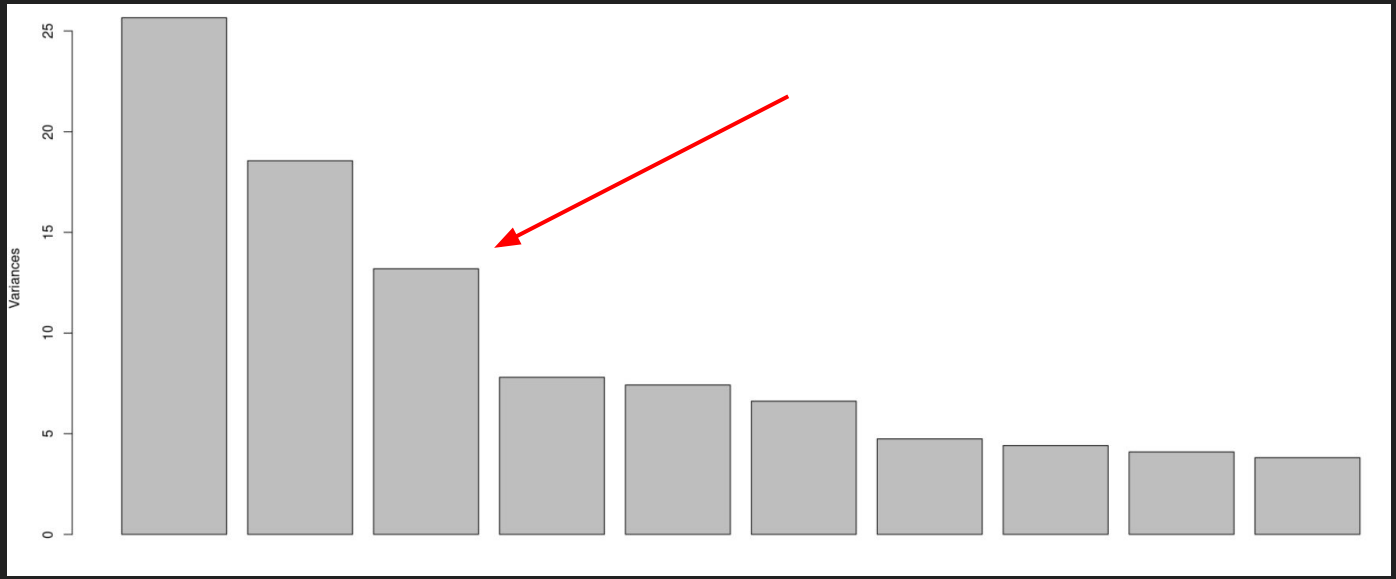
lm(formula = rw ~ feature_0 + feature_5 + feature_6
+ feature_11 + feature_12 + feature_39 + feature_46
+ feature_51 + feature_89, data = train_scale)

← Most relevant feature model

```
Residual standard error: 0.9831 on 2427 degrees of freedom
Multiple R-squared:  0.03699,   Adjusted R-squared:  0.03342
F-statistic: 10.36 on 9 and 2427 DF,  p-value: 7.763e-16
```

PCA

Optimal number of principal components ~ 3

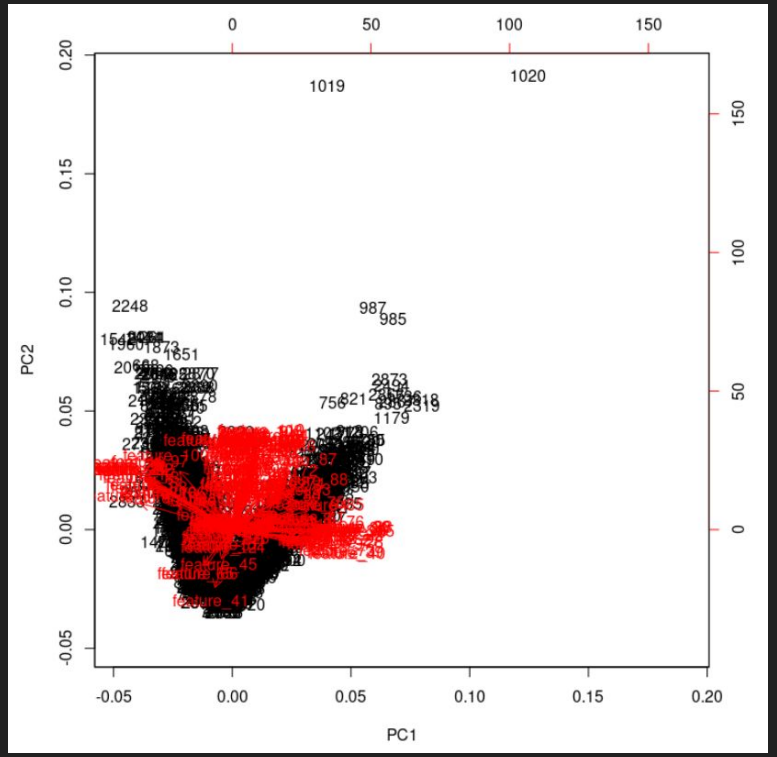


EUREKA!

Cluster groupings were observed among feature entries

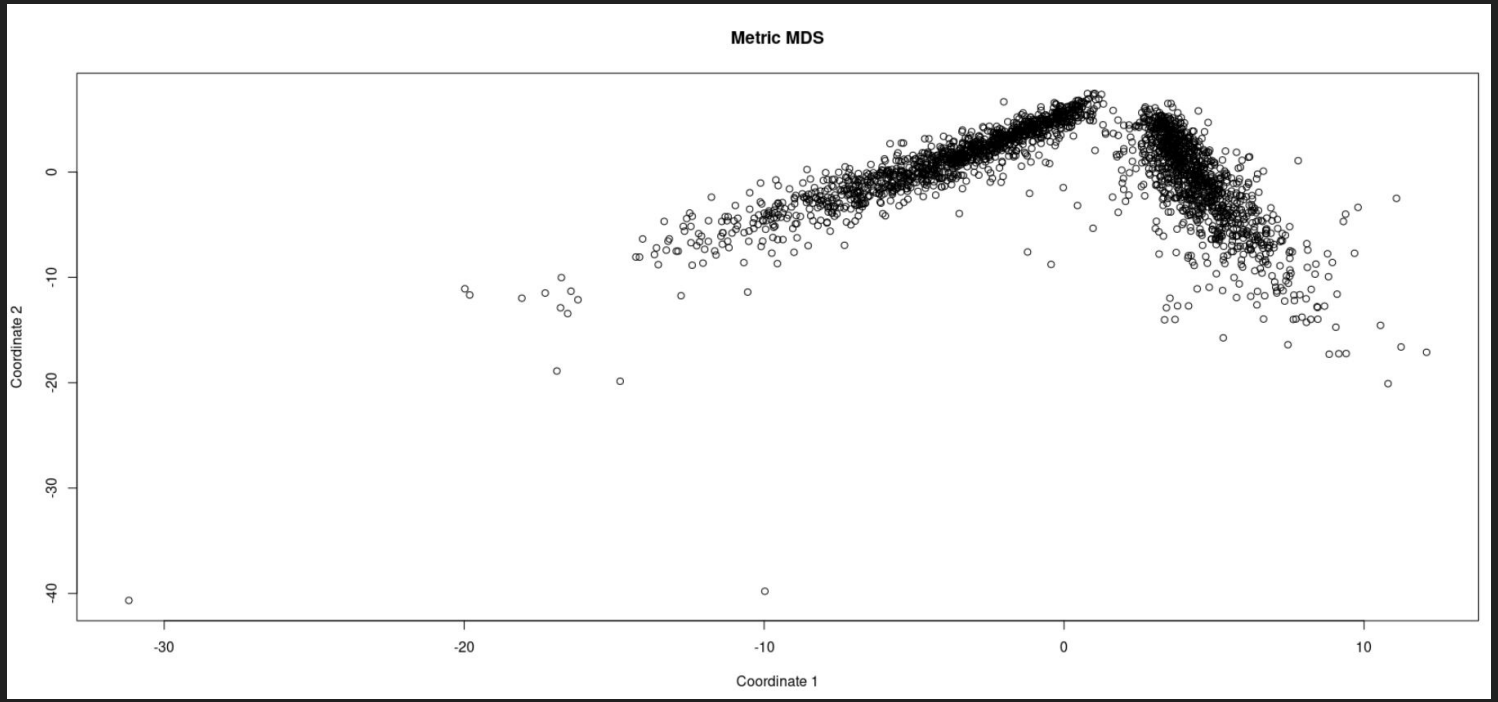
```
Mean item complexity =  1.2
Test of the hypothesis that 2 components are sufficient.

The root mean square of the residuals (RMSR) is  0.15
 with the empirical chi square  955869.3  with prob <  0

Fit based upon off diagonal values = 0.7
```

```
Mean item complexity =  1.4
Test of the hypothesis that 3 components are sufficient.

The root mean square of the residuals (RMSR) is  0.12
 with the empirical chi square  571368.1  with prob <  0

Fit based upon off diagonal values = 0.82
```

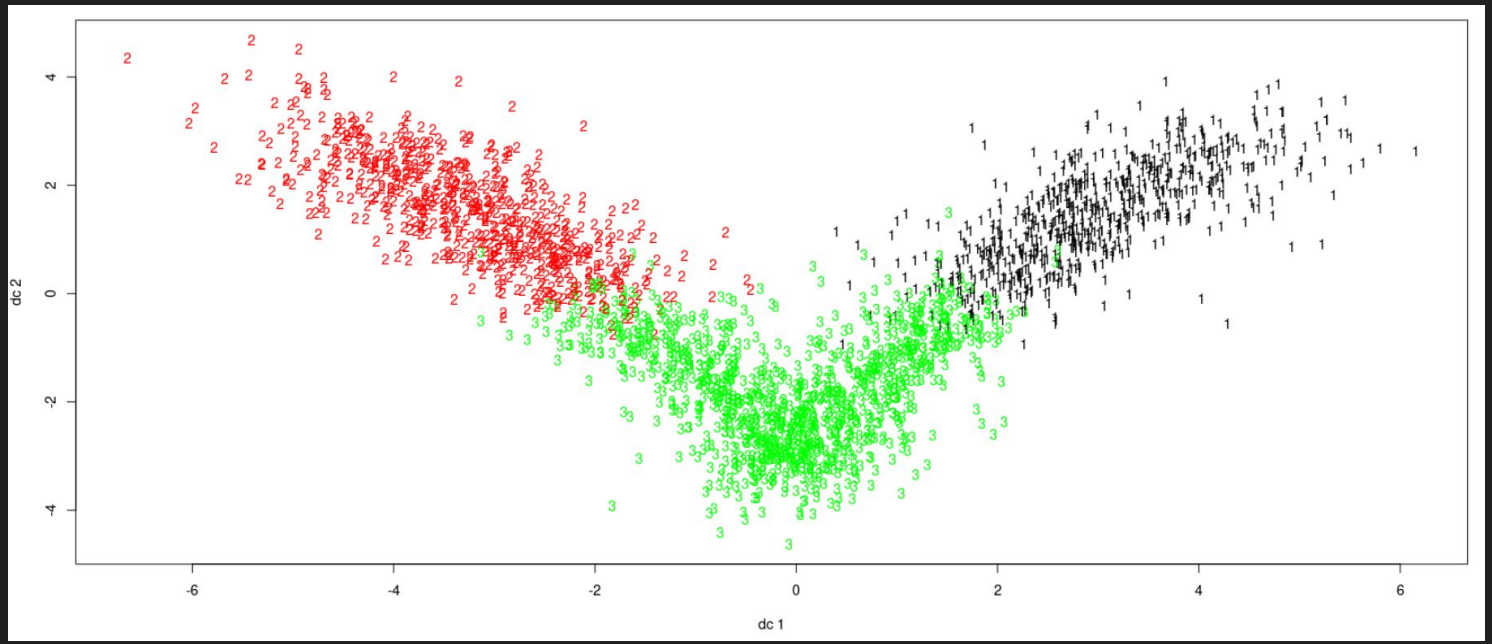


MDS - clustering pattern verified

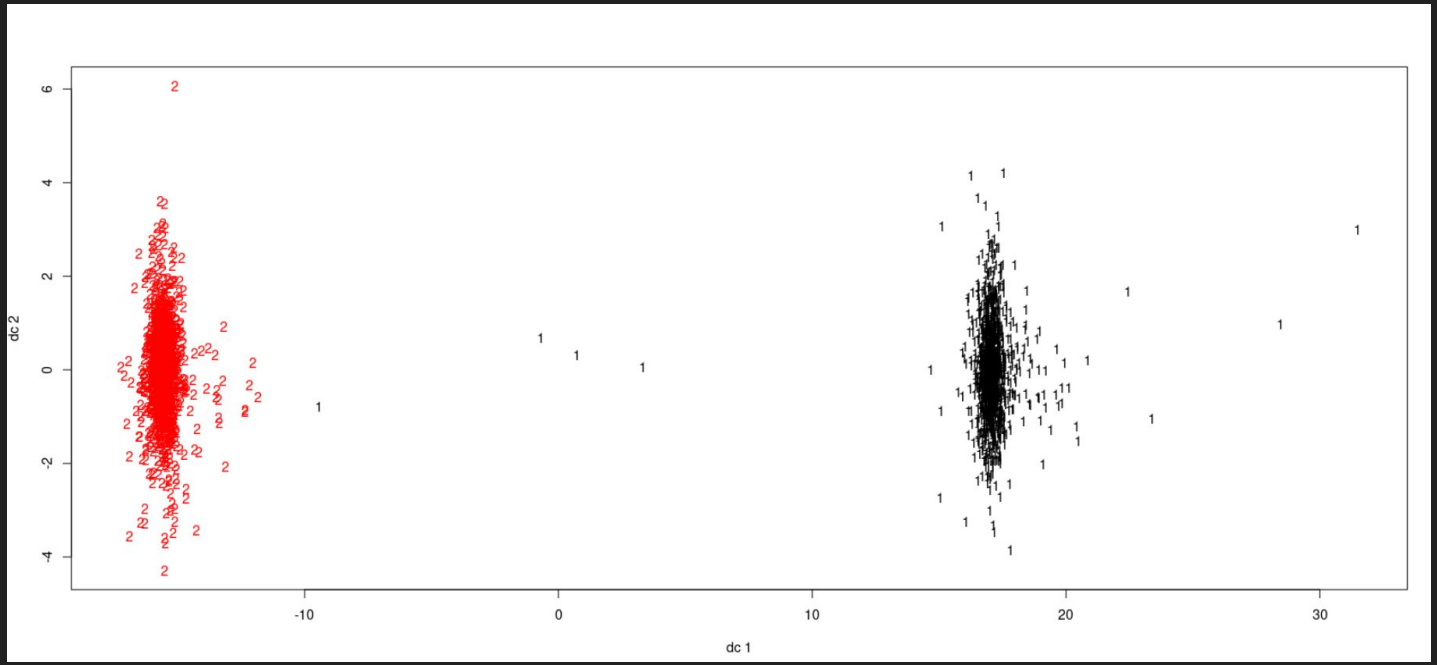
Limitation: MDS poor performance with large dataset



Clustering property verified with cluster analysis



Cluster Property with Cluster Analysis continued...



K-Means Cluster Modeling

50/50 Train / Test Split

- RW variable was discretized into a categorical variable to signify each cluster

```
data_test$rw = cut(x=data_test$rw, breaks = c(-11,2.75,7.1), labels = c(1,2))
```

Note:

2.75 was used instead of zero to signify a more confident investment opportunity

K-Means Cluster Modeling

```
>   cluster.stats(d, km1$cluster, km2$cluster)   #compare cluster fit 1 & 2
$n
[1] 2437

$cluster.number
[1] 2

$cluster.size
[1] 1161 1276

$min.cluster.size
[1] 1161

$noisen
[1] 0

$diameter
[1] 81.77554 82.87380

$average.distance
[1] 13.88533 13.92038

$median.distance
[1] 12.96156 13.10854

$separation
[1] 4.616035 4.616035
```

The goal was to identify *confident* profitable entries (RW > 2.75)

- NClusters = 2 was chosen for a broader application purpose

Clustering: Predictions with KMeans

er = mean(data_test$rw != preds)

47% error

ac = mean(data_test$rw == preds)

52% Accurate!

Limitations

-Anonymized features

-NA values - could have been filled in with medians

-Multidimensional Scaling performance

Conclusions and Future Study

Even after limitations, we were able to find patterns and validate a clustering analysis that was useful in identifying weighted responses (RW).

As there isn't a foolproof market prediction technique, this clustering technique may be used to predict valid opportunities, provided that similar feature signals are provided.