UFC Data Mining Project

Data Cleaning and Imputation, SVD

Data Cleaning and Imputation

- Filter out instances with missing stances or referee information
- Impute missing data using median for numerical values
- Verification

SVD - Basics

- Relationship to PCA
  - Let R be a matrix with rows being users and columns being ratings for fighters
  - R^T is therefore a matrix with rows being fighters and columns being ratings that users gave the fighter
  - If we were to do PCA on R and R^T, we could use it to find similar users and fighters respectively
  - Let PCA(R) = U and PCA(R^T) = M, then we have R = M∑U
- Problem: R is sparse
  - Cannot calculate eigenvectors corresponding to RR^T and R^TR
  - Thus we must solve using by minimizing:

$(R_{ui} - p_u \cdot q_f)^2$ where p_u are the vectors that comprise the rows of M specific to a user u and q_f are the vectors that comprise columns of U^T specific to a fighter for each R_{ui} in R

SVD - Implementation

- Function takes in a username, a list of fighters the user likes, and a list of fighters the user dislikes
- Internally represents the fighters the user likes as having been rated a 5
- Represents the fighters the user disliked as having been rated a 1
- Generates top 3 recommendations for fighters for user to consider

Future Work

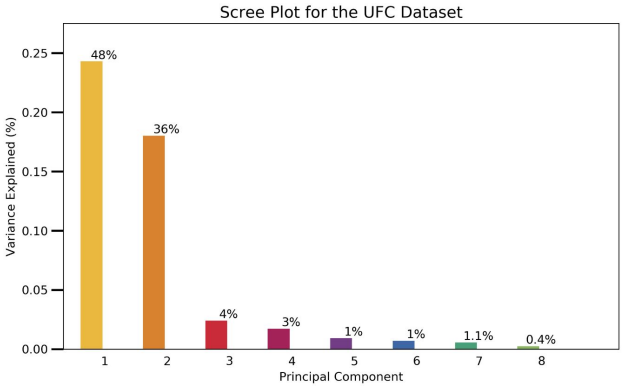
- Find way to scrape user ratings from various websites into a dataset to better demo SVD
- Find ways to more directly integrate this into a hybrid recommender model

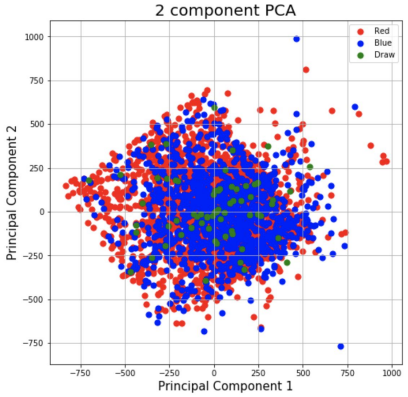
UFC Data Mining Project

PCA, K-Means and Alternative Least Squares

Principal Component Analysis (PCA)

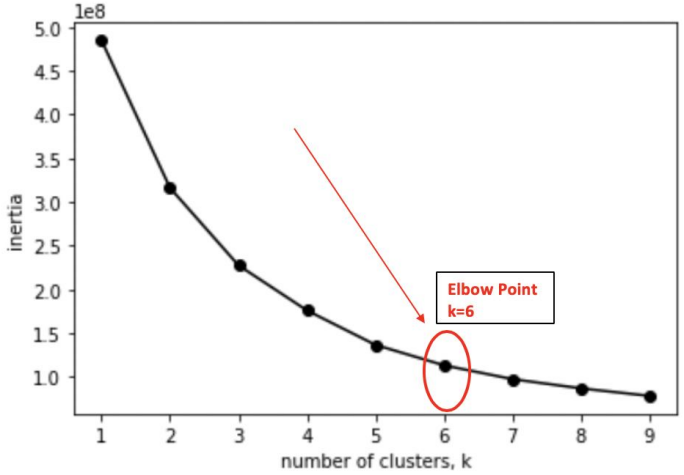
- Part of the preprocessing phase for our dataset
- Attempted to visualize our high dimensional dataset
- Dimensionality reduction to two components based on scree plot
- Incorporate PCA in K-means clustering

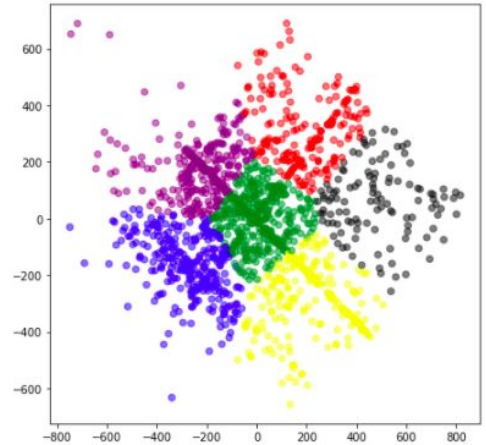




K-Means Clustering

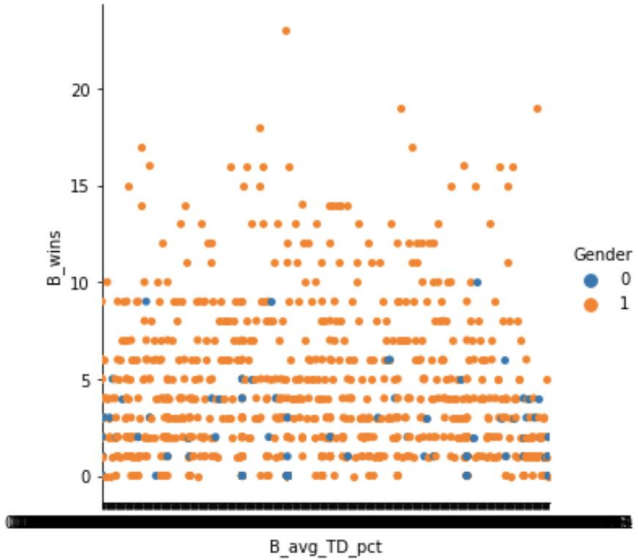
- Optimal cluster number k=6 from inertia plot
- Lack of separation of clusters and non-linear correlation of the data caused PCA and ultimately K-Means clustering to fail





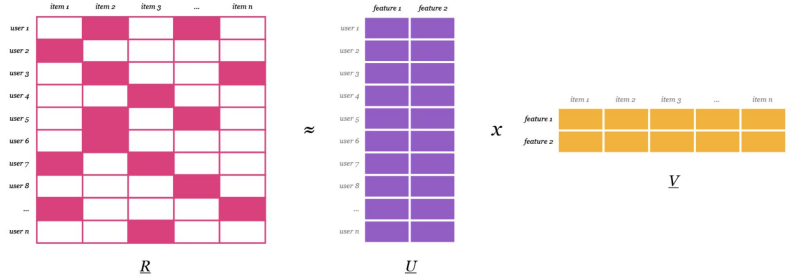
Categorical Plots

- Attempted to find common features of male and female fighters and visualize some winner-like attributes
- Female labelled as 0 Male as 1
- Female data is sparse compared to male so difficult to predict
- No clear interpretation of winning attributes



Alternating Least Squares

- Another implementation of a collaborative recommendation algorithm
- Based on the implicit nature of our data we attempt to arrive to a factorized version of our original data
- Fitting the model using a sparse feature-fighter matrix
- Goal is to find similar and top ranked fighters based on a particular feature



ALS Collaborative Filtering (2017)

Alternating Least Squares

- Data as feature-fighter and fighter-feature matrix (Feature is takedown percentages). One matrix for fitting the model and the other for recommendations.
- Implicit library is used to fit model and recommend fighters
- After fitting model
  - Find similar fighters
    - Based on takedown percentages like above
    - similar_items func provides similar fighters
  - Top fighter recommendation
    - Based on takedown percentages like above
    - Dot product of fighter vector and weight class vector
    - Scaled to provide score for each fighter
    - Result: Each fighter is assigned a score to distinguish top fighters
- Drawback
  - Single feature is used to evaluate recommendations

```
get_similar_fighters("Tony Ferguson")
```

```
Forrest Petz
Max Griffin
Chad Laprise
Chris Brennan
John Howard
Yoshiyuki Yoshida
Carlo Pedersoli
Kyle Noke
Luigi Vendramini
Josh Haynes
```

	Fighter	score
0	Ricardo Ramos	0.041794
1	Jason Reinhardt	0.030534
2	Jeff Curran	0.029447
3	Chris Gutierrez	0.027753
4	Thomas Almeida	0.027238
5	Byron Bloodworth	0.025316
6	Marcos Vinicius	0.023834
7	Walel Watson	0.021394
8	Damacio Page	0.020690
9	Nick Denis	0.020363

Future Work

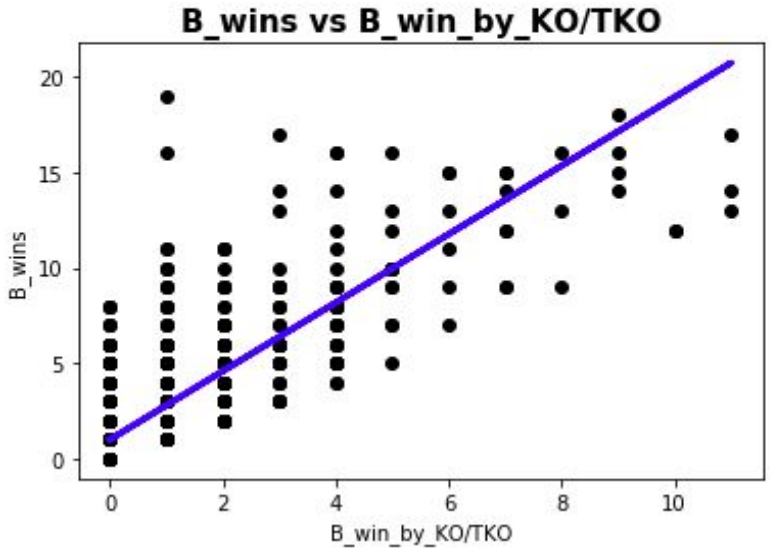
- Evaluate other recommender algorithms
- Evaluate performance of ALS using Mean Precision Analysis, Serendipity and Novelty Calculations
- NLDR possibly for data preprocessing

UFC Data Mining Project

Linear Regression and k-Nearest Neighbors

Linear Regression

```
Coefficients:
 [1.79382125]
Mean squared error: 4.27
Coefficient of determination: 0.61
Variance Score: 0.61
```



k-Nearest Neighbors

- Instance Based
  - Uses raw training data to make predictions and classifications
- Does not require a training phase
- Looks at features of a fighter
  - Checks the vicinity of the features and determines the amount of neighbors
  - Uses Minkowski distance (generalized Euclidean and Manhattan)
  - Classifies based on nearest neighbors and majority voting
- Curse of Dimensionality
  - Does not scale well with huge datasets

k-Nearest Neighbors -

For Recommending Valentina Schevchenko

```
femaleRecs = valentinaRecommendations.loc[(valentinaRecommendations['Gender'] == 0)]
femaleRecs
```

	R_fighter	B_fighter	Referee	date	location	Winner	title_bout	weight_class	no_of_rounds	B_current_lose_streak
1635	Amanda Nunes	Ronda Rousey	Herb Dean	2016-12-30	Las Vegas, Nevada, USA	Red	True	Women's Bantamweight	5	1.0

1 rows × 146 columns

References

ALS Implicit Collaborative Filtering - Rn Engineering

Building a Collaborative Filtering Recommender System with ClickStream Data

Recommender Systems — It's Not All About the Accuracy