Drafting Success: Using Machine Learning to Identify Star Quarterbacks

By: Jimmy Brawner, Michael Hargroder, Nathan Kosiba, and Chris Marchini

5/9/2020

Key Objective

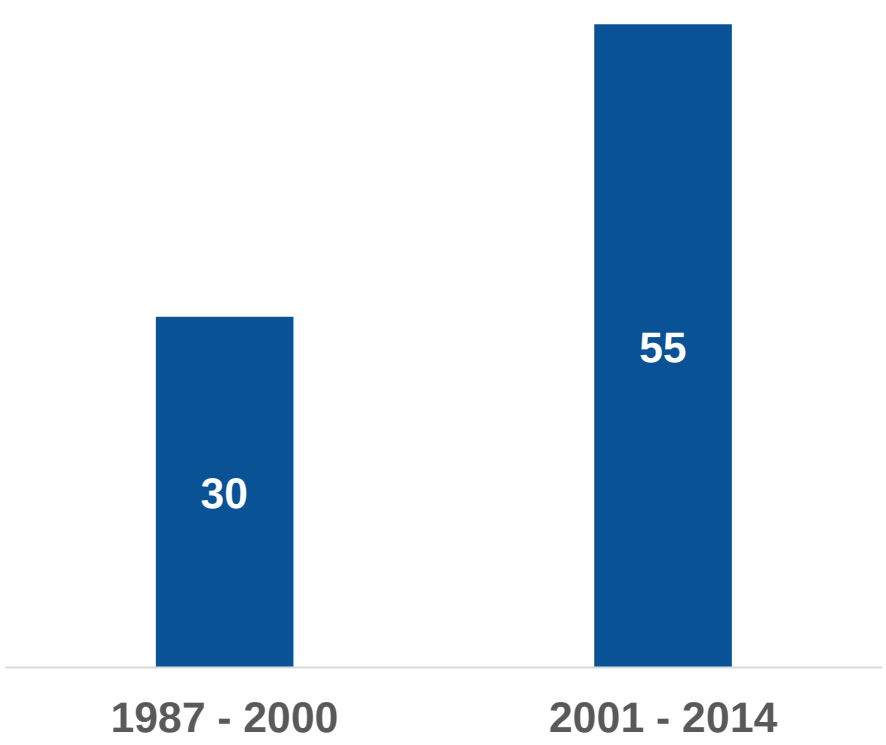
Using a variety of **extant data sources**, we **leveraged** Machine Learning techniques to build a model that **predicts success** of quarterbacks from the recent NFL draft class.

Why look at NFL draft picks?

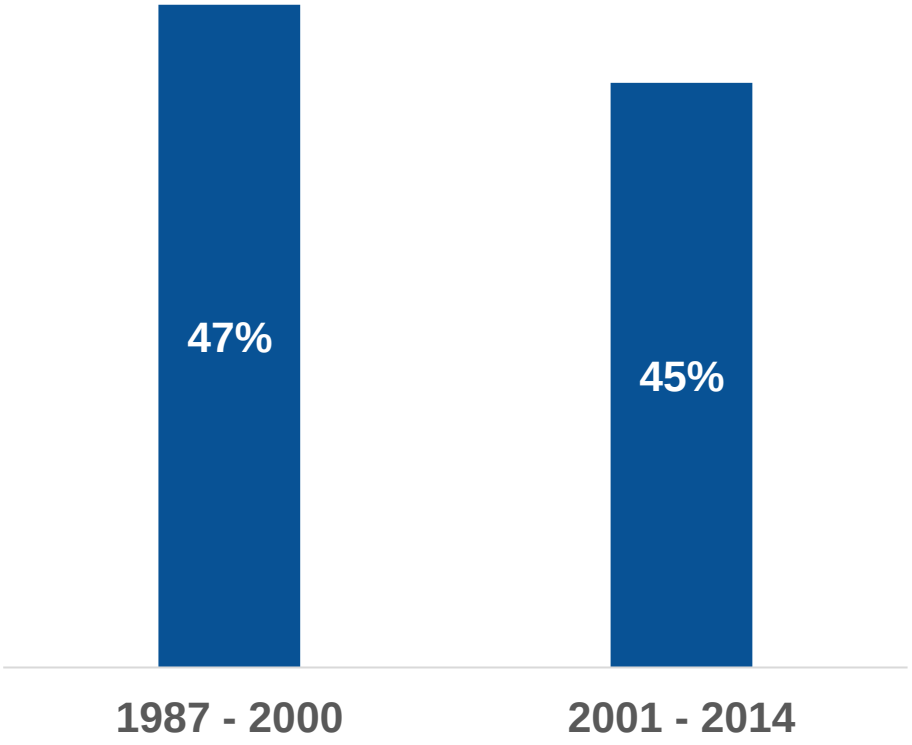
- *Moneyball*-ification of sports, FiveThirtyEight
- Determine if predicting success of a draftee is feasible
- Demystify data analysis in sports
- Draft picks can make or break a team
- Teams are not great at predicting QB success

QBs drafted in 1st and 2nd round since 1987

GMs have been drafting more QBs early …



… but QB success rates have stayed almost flat



Source: Sports-reference.com. Sample size measured was 85 QBs for 1987 – 2014

Enter our model ...

- Data used for players:
  - College football stats
  - NFL combine metrics
  - Personal biometrics

- Data sources used:
  - sports-reference.com (pro and college football)
  - nflcombineresults.com

Defining a “successful” NFL QB

- Original definition
  1. At least one (1) pro-bowl appearance
  2. Made the All-Pro team
  3. Made the Hall of Fame
  4. Started for at least six (6) seasons

- Relaxed definition for model:
  1. At least one (1) pro-bowl appearance
  2. Started for at least five (5) seasons

The data we had access to includes …

College Stats

- Rushing Average
- Adjusted Yards/Attempt
- Efficiency Rating
- Plays/game
- Completion Percentage

NFL Combine Metrics

- 40-Yard Dash
- Vertical Leap
- Broad Jump
- Shuttle
- 3-Cone

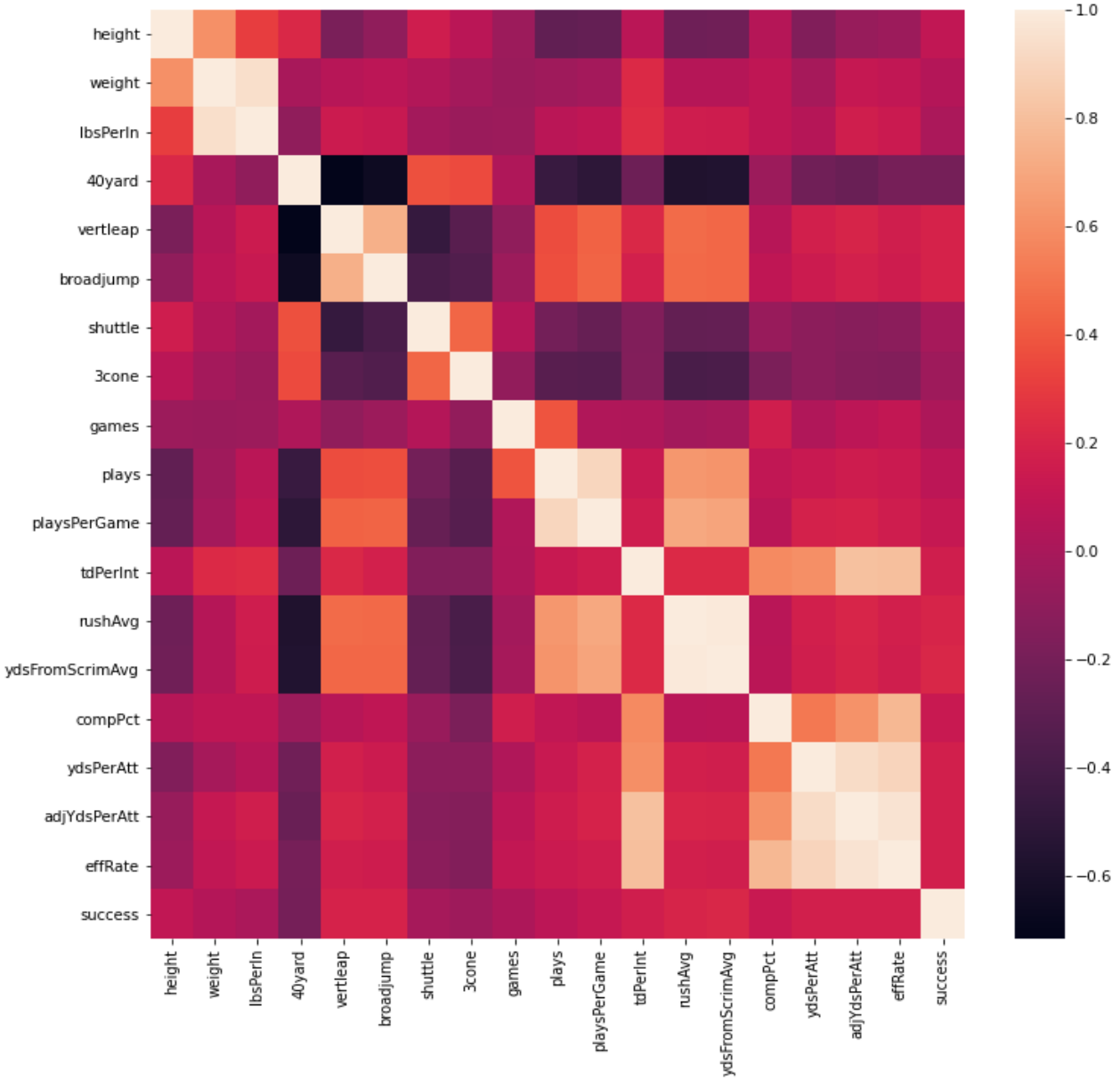
Personal Biometrics

- Height
- Weight

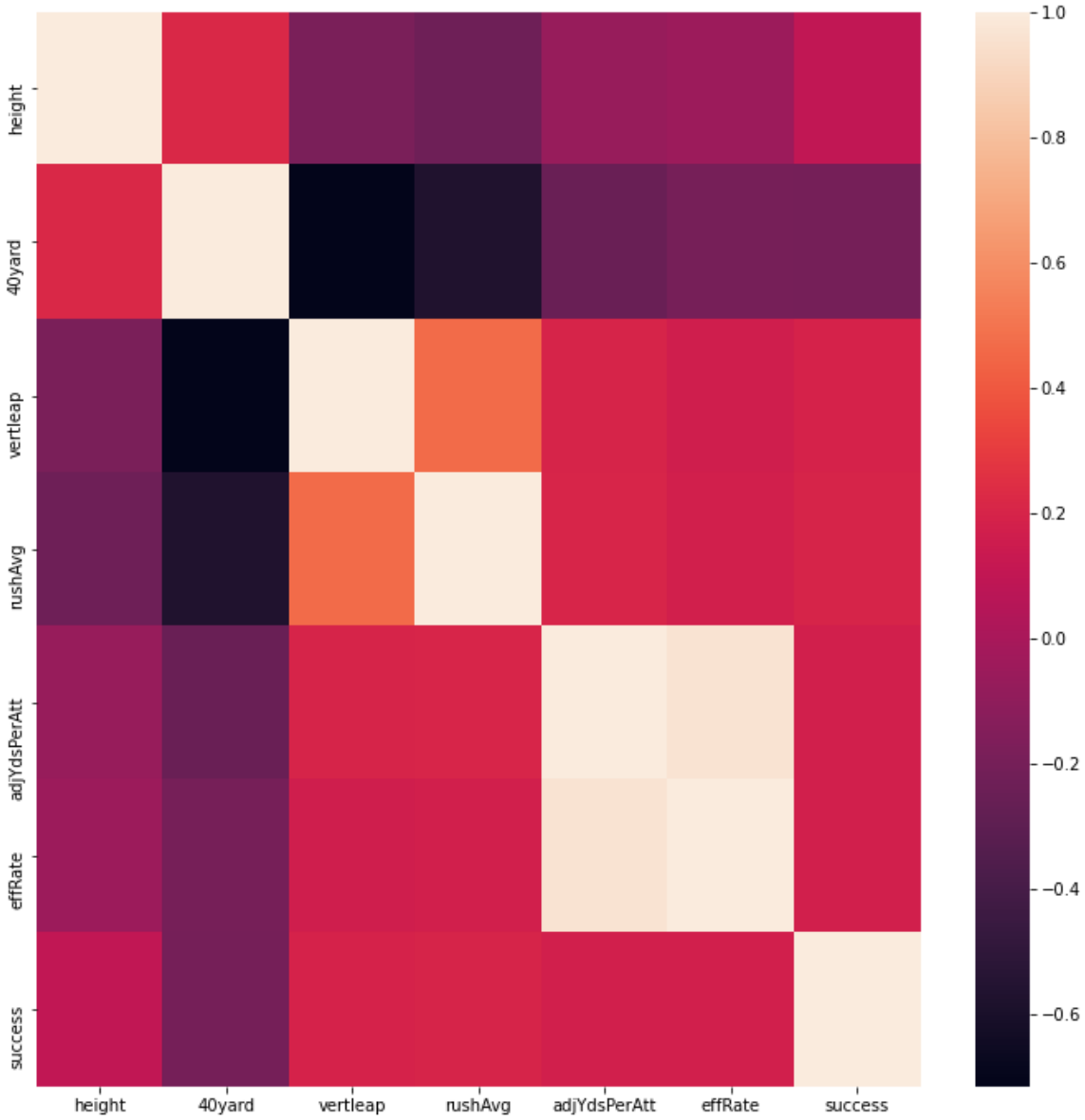
Our ETL Process

1. Used Pandas to work with our data
2. Filled missing combine data using mean values
3. Used Seaborn (Matplotlib) to visually explore our data
4. Loaded individual tables into SQLite database
5. Created model data from merging tables from our SQLite database

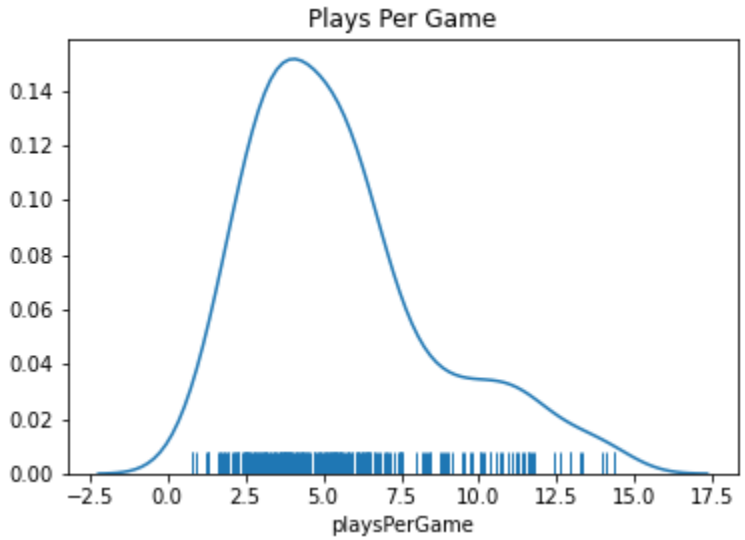
Correlation of all features

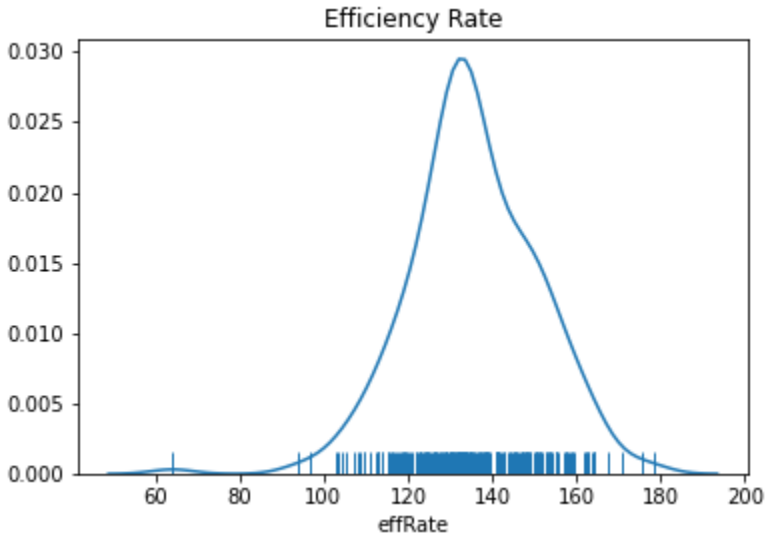


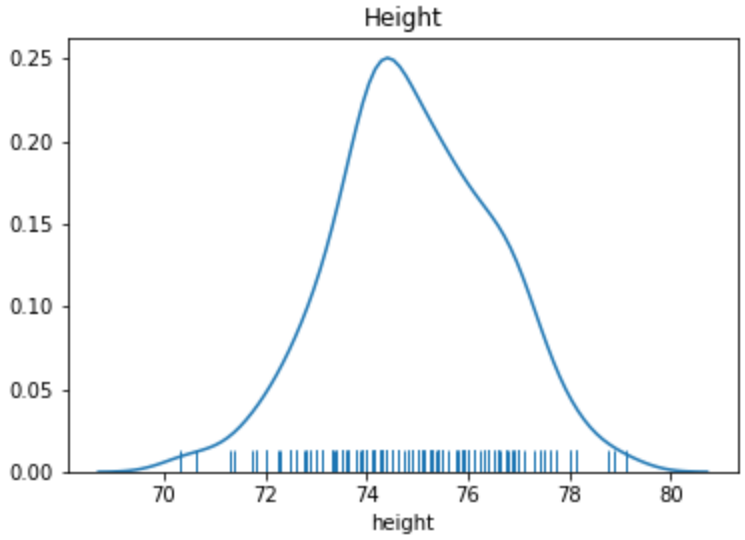
Correlation of narrowed-down features

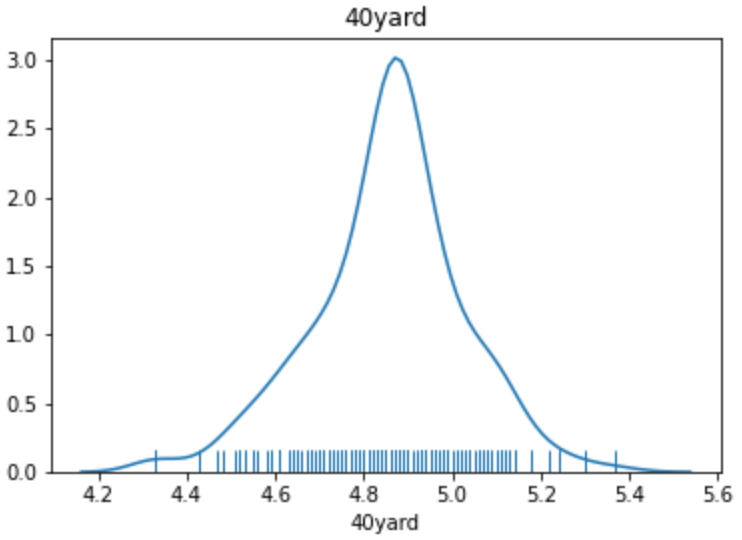


Narrowed-down feature distributions









Our smoothed features post-ETL

College Stats

- **Rushing Average**
- **Adjusted Yards/Attempt**
- **Efficiency Rating**
- ~~Plays/game~~
- ~~Completion Percentage~~

NFL Combine Metrics

- **40-Yard Dash**
- **Vertical Leap**
- ~~Broad Jump~~
- ~~Shuttle~~
- ~~3-Cone~~

Personal Biometrics

- **Height**
- ~~Weight~~

Building our model with TensorFlow

- Made two separate test models:
  - College football
  - Combine stats
- Garbage in, garbage out
  - Early models skewed toward failures
  - Adjusted success distribution in training dataset
- Built neural network
  - Used consistent scaling of X data
  - Sequential model
  - Layers: Four (4) dense, one (1) input, two (2) intermediate, one (1) output
  - Activation function: ReLU
  - Optimization: Adam
  - Loss: Categorical Cross Entropy
  - Epochs: 1,000

Model demo

Results Summary, Why this Matters, and Next Steps

- No turnkey solution: making a reliable model is not easy
- Had a 78% success rate between 1987 and 2014
- Some variables aren't available (e.g., player intelligence)
- Can't predict unexpected events like player injuries
- We could optimize our model further

Thank you! Questions?