Using logistic regression to win fantasy football leagues

presented to: The New York Python Meetup Group

presented by:
Alain Ledon - Professor, Baruch MFE
Amit Bhattacharyya - Data Scientist, Annalect

Sep 2014

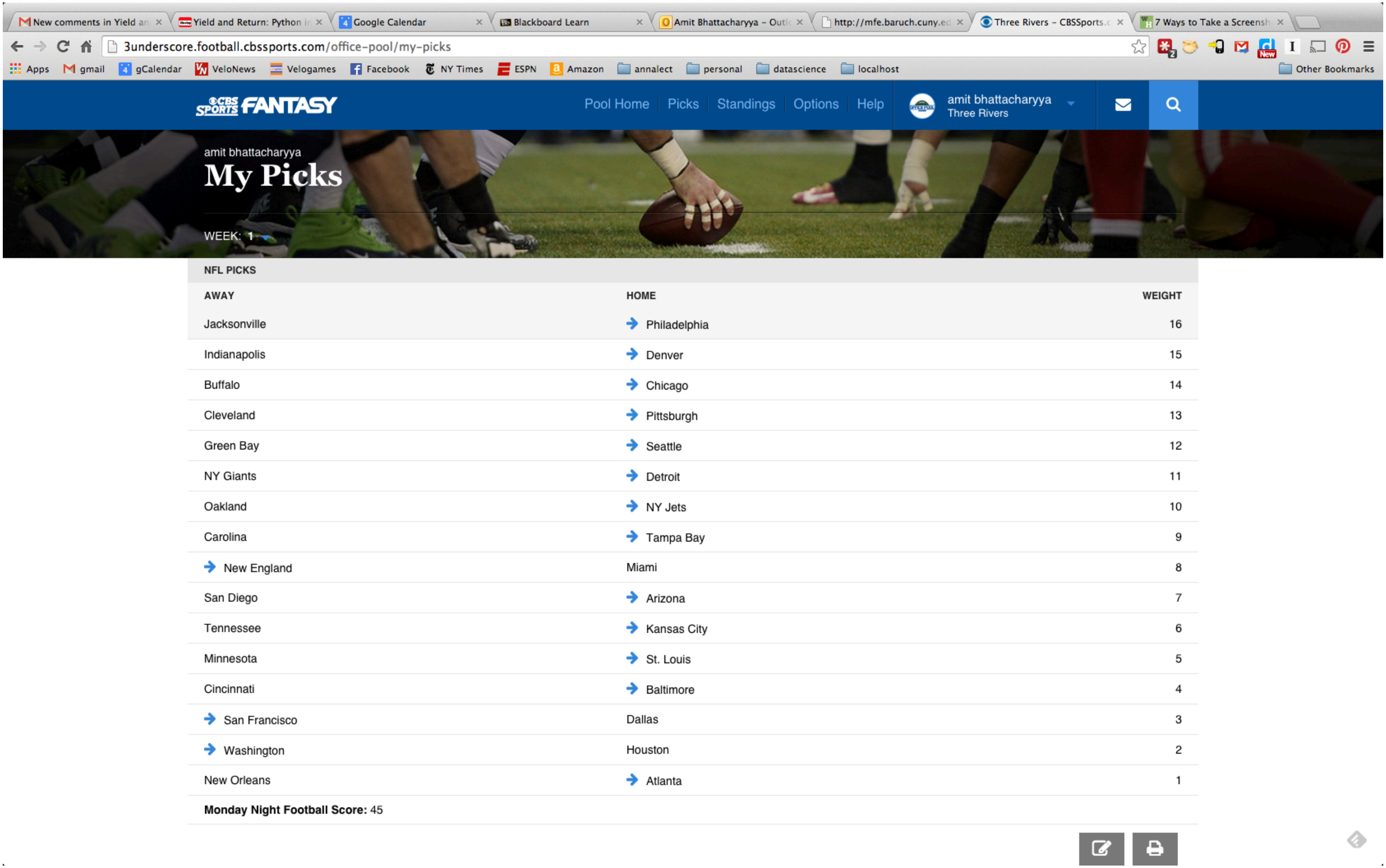
Winning fantasy football leagues

- *Doesn't every sports fan think they know how and why things happen?*
- *Are the pundits really better at predicting outcomes of games?*
- *How good are point spreads at predicting the winning teams on a consistent basis?*

While point spreads are the simplest and best starting point for winning a simple weekly "pick-em" league, application of machine learning to predict outcomes can show substantial and consistent improvement.

Fantasy league logistics

- One popular format of “pick-em” fantasy leagues in the NFL is to pick outright winners of games and rank them 1-16 based on confidence in the pick (16 highest confidence).
- Participants in the league accumulate points based on correct picks.
- The overall winner is the person with the most points at the end of season.
- Winning the league usually involves consistently picking the most likely winners and being conservative in not placing high weights on upsets.



amit bhattacharyya

My Picks

WEEK: 1

NFL PICKS

AWAY	HOME	WEIGHT
Jacksonville	→ Philadelphia	16
Indianapolis	→ Denver	15
Buffalo	→ Chicago	14
Cleveland	→ Pittsburgh	13
Green Bay	→ Seattle	12
NY Giants	→ Detroit	11
Oakland	→ NY Jets	10
Carolina	→ Tampa Bay	9
→ New England	Miami	8
San Diego	→ Arizona	7
Tennessee	→ Kansas City	6
Minnesota	→ St. Louis	5
Cincinnati	→ Baltimore	4
→ San Francisco	Dallas	3
→ Washington	Houston	2
New Orleans	→ Atlanta	1

Monday Night Football Score: 45

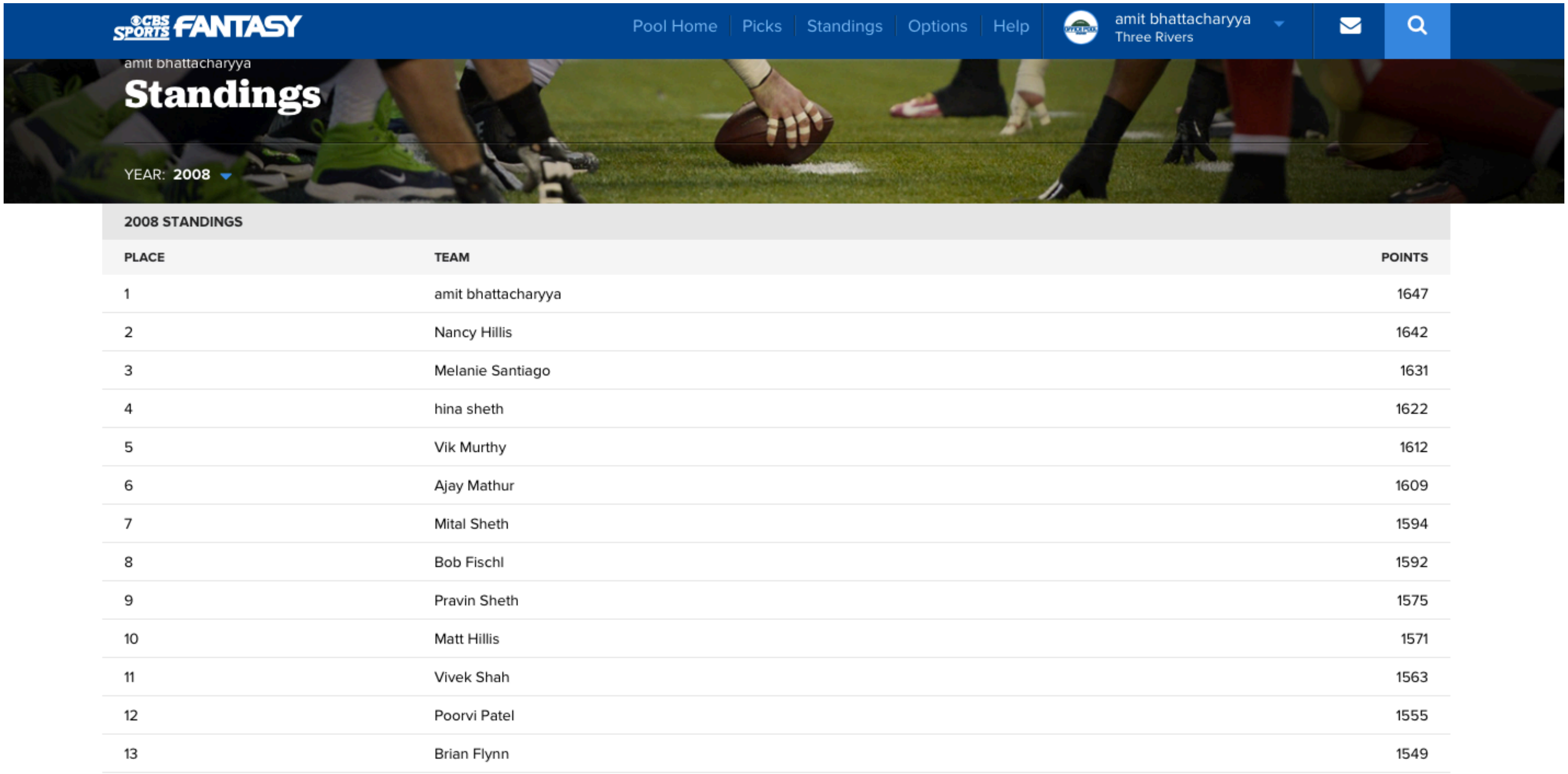
Fantasy league strategies

- The betting world already provides a view on the likelihood of a particular team winning via the point-spread. A simple strategy is to rank the choices each week by the Vegas-provided spread.
- Ad-hoc decisions based on
  - win-loss records of teams
  - opponents strength of schedule
  - home vs away game
  - division vs non-division game
  - injury reports
  - personal preferences and intuition

Ideally the point spread encapsulates all these scenarios into a single number.

How well does spread strategy work?

In 2008, the simple spread strategy would have lost by 11 points.



amit bhattacharyya

Standings

YEAR: 2008

2008 STANDINGS

PLACE	TEAM	POINTS
1	amit bhattacharyya	1647
2	Nancy Hillis	1642
3	Melanie Santiago	1631
4	hina sheth	1622
5	Vik Murthy	1612
6	Ajay Mathur	1609
7	Mital Sheth	1594
8	Bob Fischl	1592
9	Pravin Sheth	1575
10	Matt Hillis	1571
11	Vivek Shah	1563
12	Poorvi Patel	1555
13	Brian Flynn	1549

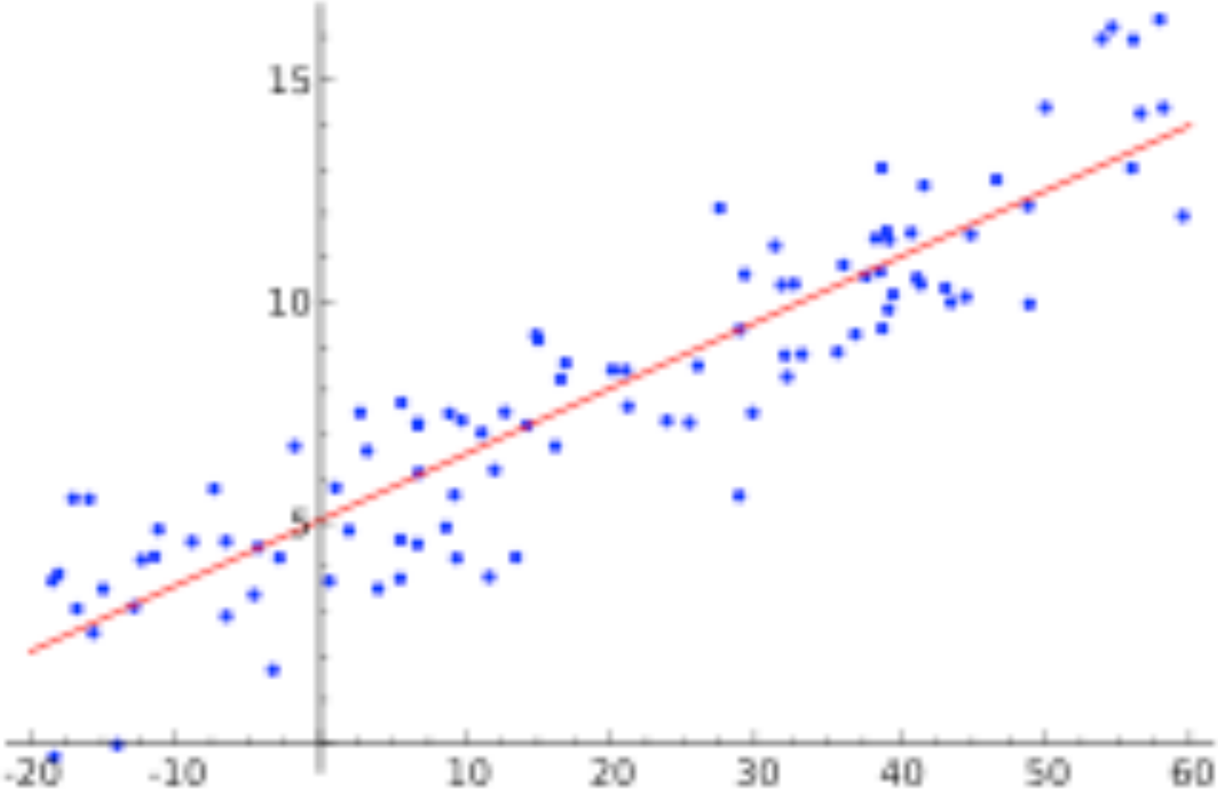
On average the spread strategy would have won in about half the years. It is important to remember that this strategy requires no thinking or guessing of any sort.

year	Winning score	Spread score	margin
2008	1647	1636	-11
2009	1710	1708	-2
2010	1590	1593	3
2011	1670	1691	21
2012	1632	1623	-9
2013	1653	1655	2

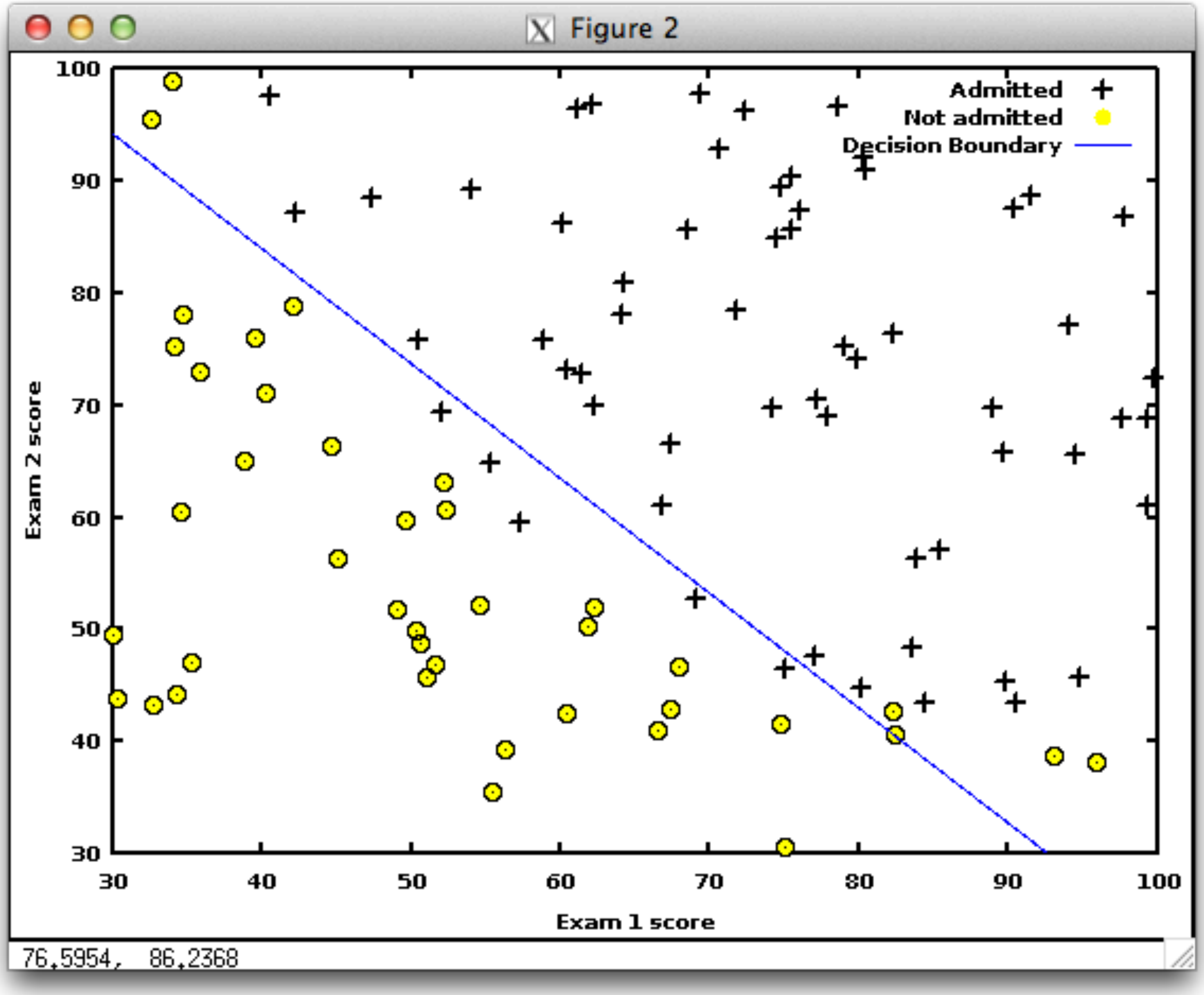
Can we do better? Can we win every year?

Linear vs logistic regression

Linear regression is used to estimate a value based on a set of x vs. y data based on a linear model.

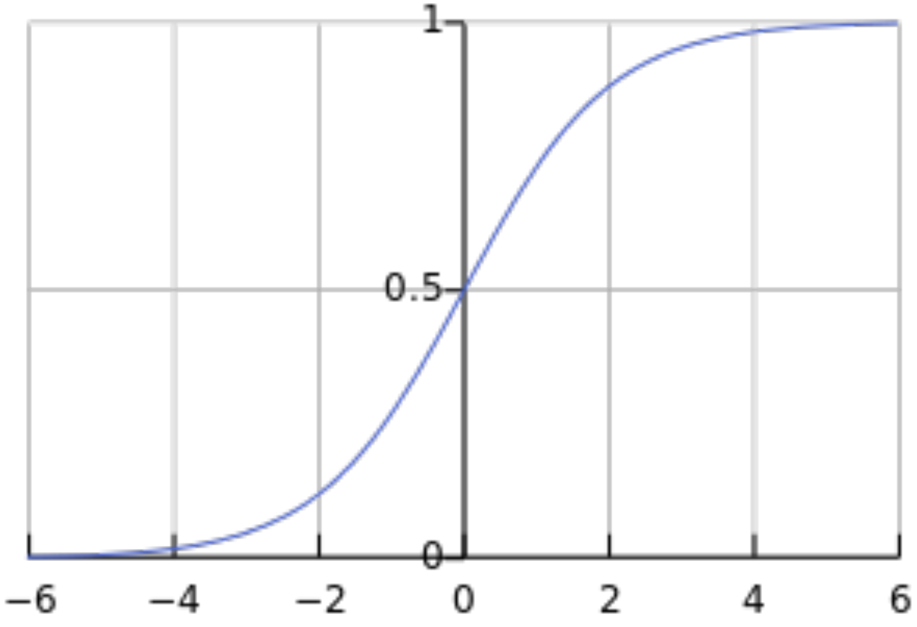


Logistic regression is used to classify data into two or more different groups. For simple cases a decision boundary can be used to visualize the classification.



The **sigmoid function** is often used in logistic regression as a binary classifier because of its non-linear shape. If the predicted value is > 0.5 then classifier is True (or 1), if < 0.5 then False (or 0).

Since the goal is to predict a binary outcome, i.e. wins and loses, it makes sense to use logistic regression for our NFL fantasy league predictions.



Using logistic regression

The simplest form of logistic regression has
- N number of inputs (called features)
- a single output for the binary classifier

For the NFL fantasy league
- need to determine the relevant features
- carefully consider what the classifier should predict

Use Python’s ***scikit-learn*** package to simplify implementation of logistic regression

http://scikit-learn.org/

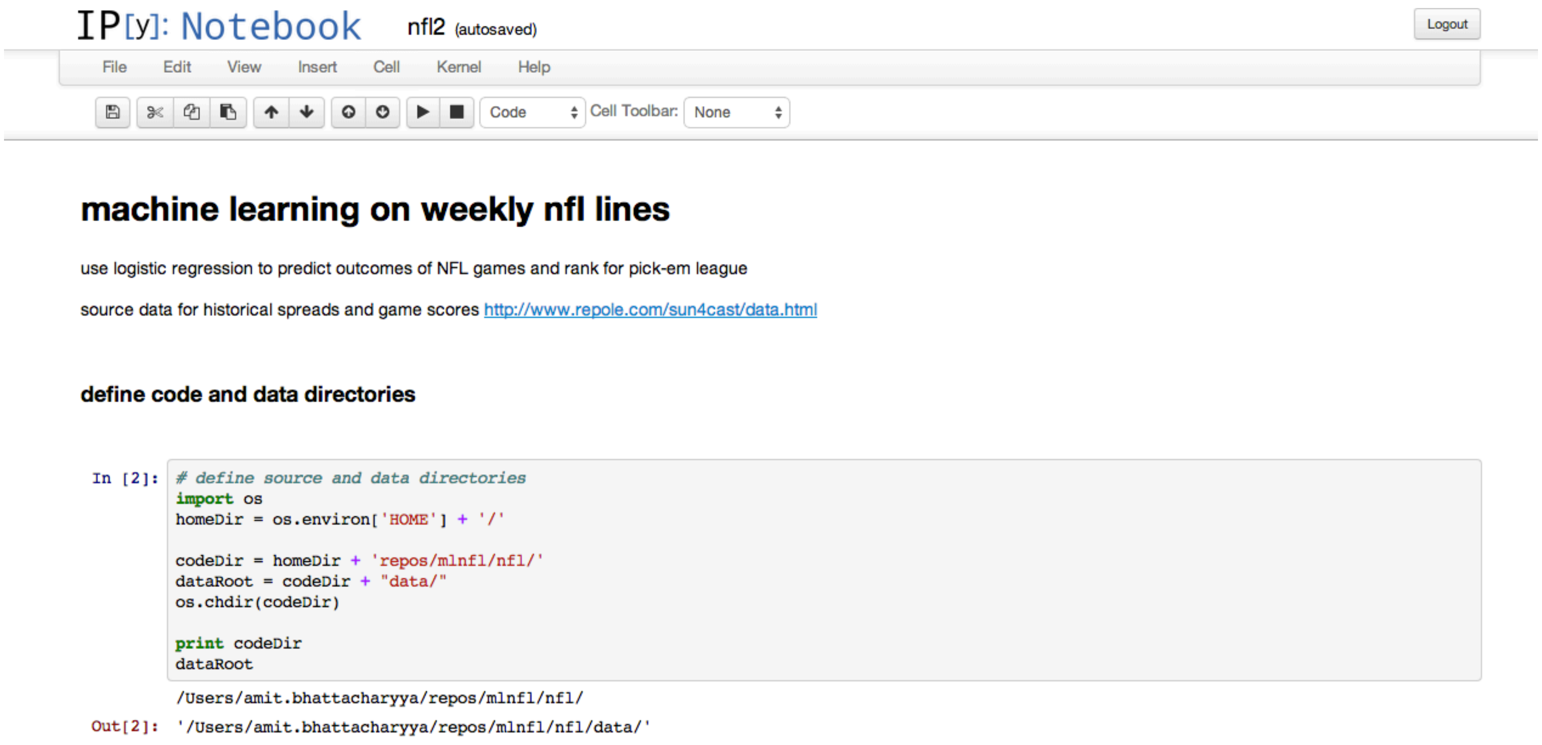
Once the features (X) and classifiers (y) are properly defined, running the logistic regression is quite simple.

```
# define model
logreg = linear_model.LogisticRegression()
logreg.fit(X, y)

# compute training accuracy
sc = logreg.score(X, y)

# get results from pre-computed logreg object
p = logreg.predict(predict_X)  # 0/1 classifier
```

Use an **iPython notebook** to interactively run Python code and examine results ...



machine learning on weekly nfl lines

use logistic regression to predict outcomes of NFL games and rank for pick-em league

source data for historical spreads and game scores http://www.repole.com/sun4cast/data.html

define code and data directories

```
In [2]: # define source and data directories
        import os
        homeDir = os.environ['HOME'] + '/'

        codeDir = homeDir + 'repos/mlnfl/nfl/'
        dataRoot = codeDir + "data/"
        os.chdir(codeDir)

        print codeDir
        dataRoot

        /Users/amit.bhattacharyya/repos/mlnfl/nfl/
Out[2]: '/Users/amit.bhattacharyya/repos/mlnfl/nfl/data/'
```

Does logistic regression outperform spread method consistently?

training set	testing set	spread	strat-Merckx	strat-Hinault	strat-Indurain
2013	2008	-11	90	71	77
2013	2009	-2	19	25	28
2013	2010	3	42	30	49
2013	2011	21	8	4	5
2013	2012	-9	4	42	15
2013	2013	2	47	77	56
average		**0.7**	**35.0**	**41.5**	**38.3**

Table shows values of how each strategy would perform compared to that year's winner.

3 slightly different strategies of ranking the weekly picks.

- Merckx = pick favored team, rank by probability of win
- Hinault = pick predicted team, rank by probability of win
- Indurain = pick predicted team, rank by abs(probability - .5)