Research Class 2

December 16, 2017

The materials covered in this class are:

1. Literature review: who has done similar research about March Madness before?
2. Literature review: methodology.

1 Literature Review: March Madness Research

1.1 LRMC: Prof. Joel Sokol @ Georgia Tech

LRMC stands for logistic regression Markov chain. The ranking system is based on a two step process as follows:

- Set up a Markov chain and estimate the transition probability using the regular season head-to-head scores.
- Solve the steady-state probability of the Markov chain and rank the teams according to it.

The key of the first step is to identify the probability r_x that a team is better than the other given the point difference is x. There are a few variations of methods to calibrating the transition probability.

- Fit a logistic regression equation to model the r_x as a function of the point difference x.
- Bayesian model: save for the next a few sessions.

1.2 BracketOdds: Prof. Sheldon Jacobson @ UIUC

Prof. Jacobson studies the bracket project in a probabilistic way. His method mainly focus on the probability a specific seed winning the other one. It requires some knowledge about simulation.

2 Literature Review: Methodology

2.1 Regression

2.1.1 Linear Regression

Linear regression is used to model a linear relationship between two variables. We call the data, which we know from observation, as the independent variable, and the response, or the result we want to see, as the dependent variable.

The technique that is most commonly used is called the "least squares regression", which aims to minimize the squared error of the fitting.

Table 11-1 Oxygen and Hydrocarbon Levels

Observation Number	Hydrocarbon Level x(%)	Purity y(%)
1	0.99	90.01
2	1.02	89.05
3	1.15	91.43
4	1.29	93.74
5	1.46	96.73
6	1.36	94.45
7	0.87	87.59
8	1.23	91.77
9	1.55	99.42
10	1.40	93.65
11	1.19	93.54
12	1.15	92.52
13	0.98	90.56
14	1.01	89.54
15	1.11	89.85
16	1.20	90.39
17	1.26	93.25
18	1.32	93.41
19	1.43	94.98
20	0.95	87.33

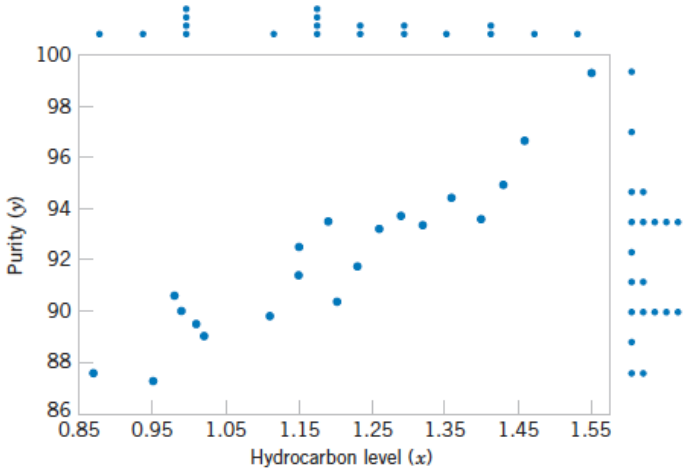

Figure 11-1 Scatter diagram of oxygen purity versus hydrocarbon level from Table 11-1.

Figure 1: An example of linear regression

Mathematically, we want to fit a model as $Y = X\beta + \epsilon$. For a general case represented in linear algebra, we have:

$$\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix},$$

$$X = \begin{pmatrix} \mathbf{x}_1^\top \\ \mathbf{x}_2^\top \\ \vdots \\ \mathbf{x}_n^\top \end{pmatrix} = \begin{pmatrix} 1 & x_{11} & \cdots & x_{1p} \\ 1 & x_{21} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & \cdots & x_{np} \end{pmatrix},$$

$$\boldsymbol{\beta} = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \end{pmatrix}, \quad \boldsymbol{\varepsilon} = \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{pmatrix}.$$

Figure 2: A representation of linear regression matrix

To minimize the squared error, we can write down the squared error as:

$$\begin{aligned} \epsilon^\top \epsilon &= (Y - X\beta)^\top (Y - X\beta) \\ &= \beta X^\top X\beta - 2\beta^\top (X^\top Y) \end{aligned}$$

According to the quadratic function's minimum, we can yield $\hat{\beta} = (X^\top X)^{-1}(X^\top Y)$. The process of deriving this in one-dimensional space is left for the exercise. The way to fit a simple linear regression in Python is to use scikit-learn

and numpy. The tutorial can be found here: `http://scikit-learn.org/stable/modules/linear_model.html`. For the example above, you can execute the following script to fit a linear regression model. The last two lines will print out the fitted coefficient and intercept.

```
import numpy
from sklearn import linear_model
x = numpy.array([0.99,1.02,1.15,1.29,1.46,1.36,0.87,1.23,1.55,1.40,1.19,
      1.15,0.98,1.01,1.11,1.20,1.26,1.32,1.43,0.95])
y = numpy.array([90.01,89.05,91.43,93.74,96.73,94.45,87.59,91.77,99.42,93.65,93.54,
      92.52,90.56,89.54,89.85,90.39,93.25,93.41,94.98,87.33])
x = x.reshape(20,1)
lr = linear_model.LinearRegression()
lr.fit(x,y)
lr.coef_
lr.intercept_
```

2.1.2 Logistic Regression

Logistic regression is to predict the binary outcome based the independent variables. The response here is the probability, which is between 0 and 1. Logistic regression model can be considered as a generalized linear model because:

$$\pi_i = Pr\{Y_i = 1|X_i = x_i\} = \frac{\exp(\beta_0 + \beta_1 x_i)}{1 + \exp(\beta_0 + \beta_1 x_i)}$$

or

$$\log\left(\frac{\pi_i}{1-\pi_i}\right) = \beta_0 + \beta_1 x_i$$

Similarly, logistic regression can also be carried out in Python using scikit-learn. Suppose we already know a dataset X and binary responses y.

```
clf = linear_model.LogisticRegression()
clf.fit(X, y)
```

2.2 Markov Chain

Markov chain is a stochastic model describing a sequence of possible events in which the probability of each event depends only on the state attained in the previous event.

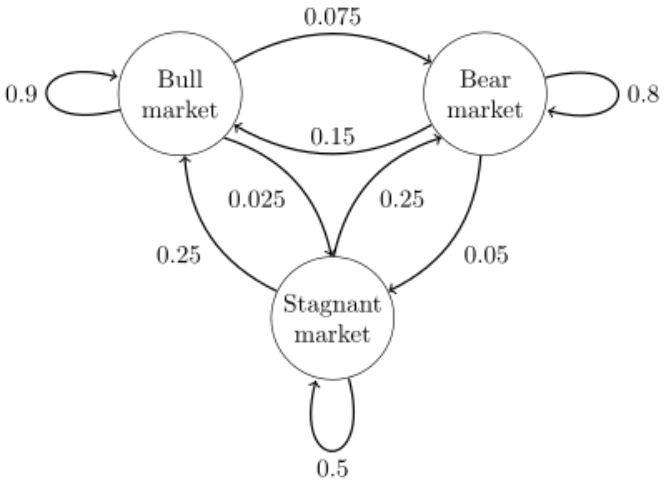

Figure 3: An example of a discrete time Markov chain