Project Organization:

* Function based programming; execution code is very short while function code is substantial
* Function based decisions on size of function re: use for other things
* All one script for funcs; one for execute
  + How to organize script files?
    - Each function is a distinct script file 🡪 one script for all functions?
    - Each step of execution code is a distinct script 🡪 one script for whole process?
      * Def not one for whole process since only one data load needs to happen for the whole validation process, but calling multiple scripts to accomplish a task introduces more opportunities for human error…
* How many lines until a function ‘should’ be broken into multiple functions?
  + Keeping in mind I hope to have others helping implement other sports, using the code.

Model validation job processing:

* Job batching for large scale model validation
  + Amazon compute cloud?
    - [www.aws.amazon.com/ec2/#details](http://www.aws.amazon.com/ec2/#details)
      * Two ways to get service
        + Pay for face value; it’s yours
        + Continuous auctions for unused capacity

Get instance paying very little

As soon as someone wants to pay full price the instance closes down

* + - Twice the cost of alternatives
      * Phoenix nap
        + Flat rate

Data cleaning:

* Some features are only included for later years data
  + For now just excluding all features that are missing for any year 2001-2011
  + Training set already happened; what matters is predictions in 2013; data is not missing in 2013…how well you do with predictions in the absence of the data
    - Gbm – Surrogate splitting allows missing data
      * Third stem from node; is bigger than x, is smaller than x, is missing
    - For methods that do not support missing values, read in to PCA with missingness
* Include closing point spread?
  + For now, yes
  + Fit prediction rule with closing line
    - Table with juciness as a function of the line
    - Prediction rule is a rule about outcomes based on the kickoff line
    - Time series of line during the week

Feature generation:

* K week averages; some values of k do not exist for some choices of weeks for a given prediction rule
  + Say one rule for weeks 4-7; a 5 week average will not make sense for weeks 4 and 5 and not make sense for week 6 games for teams that already had a bye
    - For now, excluding features that are NA for any week in the prediction rule range of weeks

Dimension reduction

* PCA
  + Exclude response?
  + Perform PCA/dimension reduction for each ‘roll’ of the data? OR perform for the entire data set (bias?) OR perform for entire range of data to be used in validation?
  + Cutoff for component inclusion? X% of maximum score or?
  + Don’t exlude raw features; add PCA features to model matrix
    - Can make choice to exclude some/all components/raw data at a later stage

Model setup

* Generate distinct prediction rules for different ranges of weeks; quarter season rules vs half season rules vs week specific rules, etc.

External model parameters

* Number of previous weeks/seasons to include in training for each week predictions
* First season/week to use in training data? How far back to go?
  + Currently 2001 is earliest used b/c many features are not avail for pre-2001
    - Fit window rarely straddles region that has data w/ and w/out a feature.
* Go back as far as possible – can make decisions about what data to pay attention to and which to ignore in making model decisions
* Choice for range of weeks for distinct prediction rules
  + Take all of the data, stratify by week/month of season, and build separate machines for each strata – completely independent rules
  + Build rule for entire season
  + Middle path: add a factor for the month as a feature, post PCA
    - But does gbm find that well enough though? Wouldn’t it need to look at the error that occurs beyond the next two leafs to notice that a split on month of season is a good first split?
  + For now: use complete pooling, one machine for the entire season

Internal model parameters

* Tuning parameters
* Error metric
* Classification then regression or just straight regression?

Week 1 prediction; don’t have to do it.