Hockey analytics

Finding good players using variable selection

Mladen Kolar (mkolar@chicagobooth.edu)

We are going to investigate data on all of the goals in the 2002–2014 seasons of the National Hockey League (NHL).

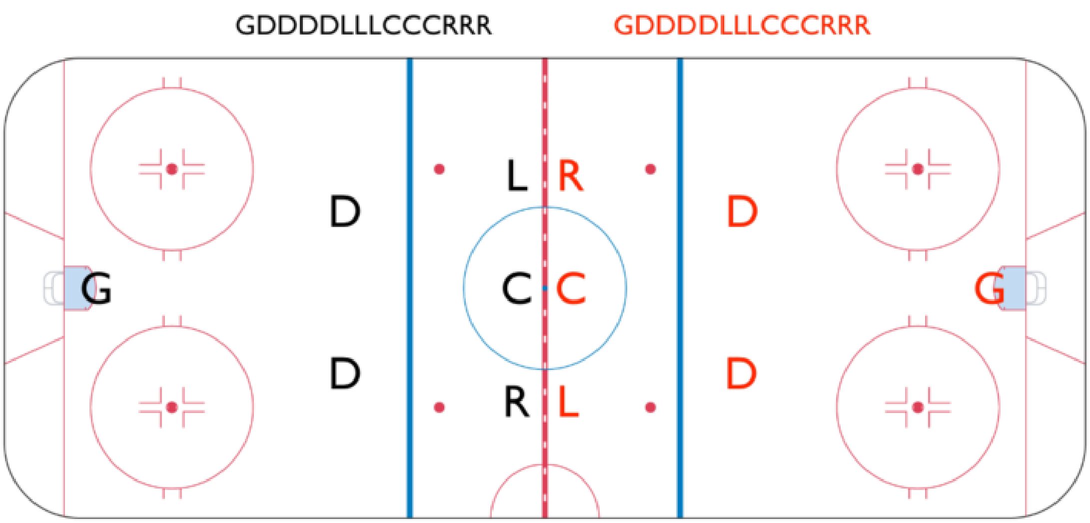
- See *Robert Gramacy, Matt Taddy, and Sen Tian*. **Hockey performance via regression**. *Handbook of Statistical Methods for Design and Analysis in Sports*, 2015. For more details.

The data is available in the `gamlr` package, a competitor of `glmnet`.

```
library(gamlr)
data(hockey)
```

- The data was scraped from NHL.com using an R package called `nhlscrapr`, spanning 11 seasons: 2002-03 through 2013–14, with playoffs.
- It includes other info we're not going to use (shots, blocked shots, penalties, etc.)

Hockey is like soccer, but on ice, 6-on-6 and with **rapid substitution**.



Quantifying player performance in hockey is hard:

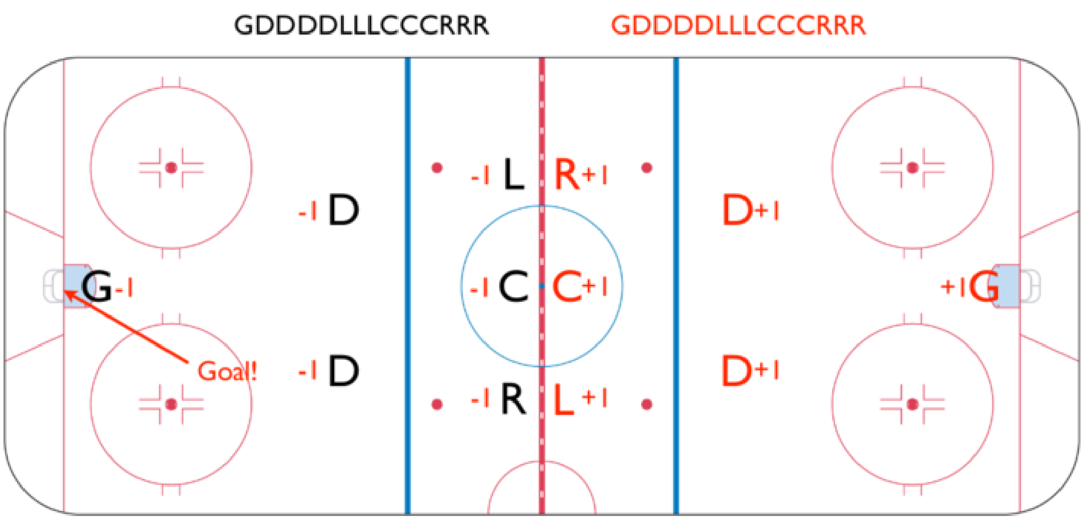
- continuous nature of play
- infrequent number of goals
- combinatorially huge numbers of player configurations.

One popular metric of individual player performance is **plus–minus**:

- the number of goals scored by the player's team,
- minus the number scored by the opposing team

while that player is on the ice.

Plus-minus is better than just goals, because it distributes the credit and blame.



The limits of this approach are obvious: there is no accounting for teammates or opponents.

In hockey, where players tend to be grouped together on "lines" and coaches will "line match" against opponents, a player's PM can be artificially inflated or deflated by the play of his opponents and peers.

In summary, two disadvantages to plus–minus:

- It is a **marginal effect**, averaging over situation, say.
- It doesn't control for sample size.

A better measure of performance would be a **partial effect**, having controlled for the effect of

- teammates,
- opponents, etc.

An appealing aspect of such an analysis is that it requires no extra data beyond that used to calculate plus–minus,

- just a (much) more involved calculation.

We will build a better performance metric with regression.

The setup

```
head(goal)
```

```
##   homegoal   season team.away team.home period differential playoffs
## 1        0 20022003       DAL       EDM      1            0        0
## 2        0 20022003       DAL       EDM      1           -1        0
## 3        1 20022003       DAL       EDM      2           -2        0
## 4        0 20022003       DAL       EDM      2           -1        0
## 5        1 20022003       DAL       EDM      3           -2        0
```

```
## 6          1 20022003        DAL        EDM       3              -1          0
```

Given n goals throughout the National Hockey League (NHL) over some specified time period, say

$$y_i = \begin{cases} +1 & \text{for a goal by the home team, and} \\ -1 & \text{for a goal by the away team.} \end{cases}$$

Then, say that

$$q_i = \mathbb{P}(Y_i = 1) = \mathbb{P}(\text{home team scored goal } i).$$

- *Home* and *away* are merely organizational devices, creating a consistent binary bifurcation for goals that can be applied across games, seasons, etc.
- Due to the symmetry in the logit transformation, player effects are unchanged when framing away team probabilities as q_i rather than $(1 - q_i)$, so we loose no generality by "privileging" home team goals in this way.

Player model

The simplest version of a model for partial player effects is the so-called **player model**, where

- the **log odds** that the home team has scored a given goal, i, becomes

$$\log\left(\frac{q_i}{1-q_i}\right) = \alpha + \beta_{h_{i_1}} + \cdots + \beta_{h_{i_6}} - \beta_{a_{i_1}} - \cdots - \beta_{a_{i_6}},$$

where

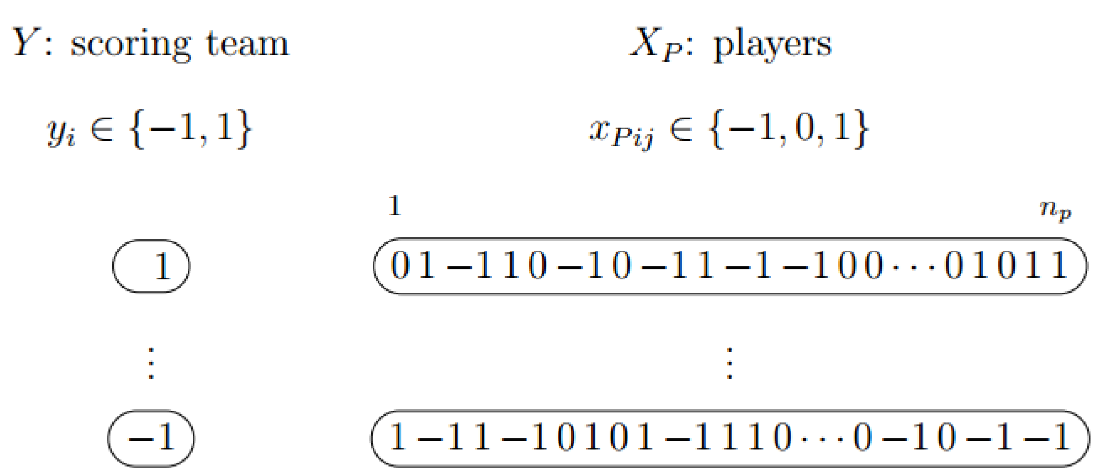
- $h_{i_1}, \ldots, h_{i_6}$ are the home team's players (i.e., player indicators), and
- $a_{i_1}, \ldots, a_{i_6}$ are the away team players.

The coefficients β_* are our **partial player effects**!

- What does α represent?

The data

How do we set up the data so that it is faithful to this format, in a logistic regression setup? Like this:



- Notice that the design matrix X_P is **sparse**.
- Sparse matrix libraries can ease storage and computational burden.

```r
player[1:3, 2:7]
```

```
## 3 x 6 sparse Matrix of class "dgCMatrix"
##      ERIC_BREWER ANSON_CARTER JASON_CHIMERA MIKE_COMRIE ULF_DAHLEN ROB_DIMAIO
## [1,]           1            .             1           .          .         -1
## [2,]           .            1             .           1         -1          .
## [3,]           .            1             .           1          .         -1
```

Getting fancier

Beyond controlling for the effect of who else is on the ice, we also want to control for things unrelated to player ability.

Embellishments abound. You can add:

- player-season indicators;
- Team or team–season indicators;
- Special teams indicators (6v5, 6v4, . . . , pulled goalie, etc.);
- Special situations: overtime, playoffs, exhibition, etc.

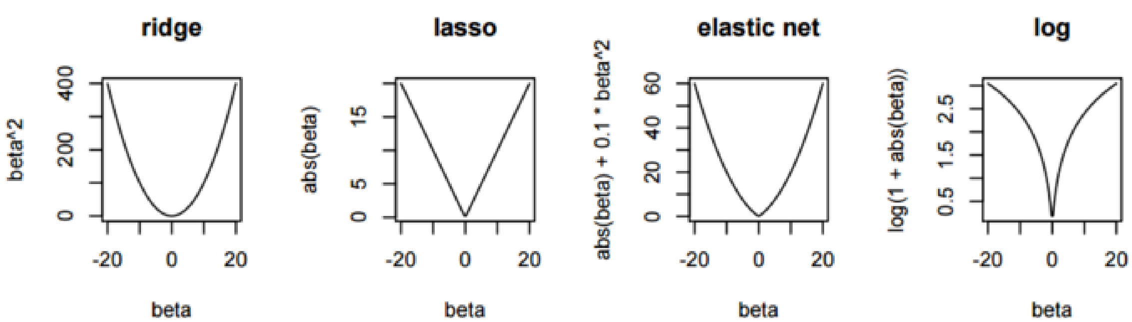
But the idea is the same:

- These are just indicator variables,
- and its all just a big logistic regression.

`gamlr` package

We will use `gamlr` package to fit the model

- `gamlr` preferrs a so-called log-gamma penaty $\text{pen}(\beta) = \log(1 + |\beta)|)$.



It works very similarly to `glmnet`, but are some small differences. E.g., it

- supports selection via information criteria, in addition to CV;
- allows some coefficients to undergo different (e.g., ridge/no) penalization.

Differential penalization

This ability to differentially penalize could be advantageous.

If (large) player partial effects (assets and liabilities) are the main interest,

- i.e., large non-zero coefficients, separating the wheat from the chaff, then it makes sense to "select" players, but be more lenient to special teams, etc.

Ok, we're sold.

Assembling the design

Lets throw everything together:

- `config` with special teams, playoff indicators, etc.
- `team` with team indicators, and
- `player` with player indicators

These are sparse matrices, so we'll need to combine them together using a new command.

```
X <- cbind(config, team, player)
y <- goal$homegoal
dim(X)
```

```
## [1] 69449  2776
```

- Woah! That's quite big.

`gamlr` call

```
nhlreg <- gamlr(X, y, free=1:(ncol(config)+ncol(team)),
                 family="binomial", standardize=FALSE)
```

- `free` denotes unpenalized columns. These are columns of the design matrix that we do not want penalized. We are using it to keep the special-teams and team-season variables unpenalized—we know that we want them in the model, and so we let them enter without restriction.
- We use `standardize=FALSE` because the columns are already indicators. This is one of the special cases where all of our penalized variables are on the same scale (player presence or absence). Without `standardize=FALSE`, we would be multiplying the penalty for each coefficient (player effect) by that player's standard deviation in the player matrix. The players with big standard deviation are guys who play a lot. Players with small standard deviation are those who play little (almost all zeros). Hence, weighting penalty by standard deviations in this case is exactly what we do not want: a bigger penalty for people with many minutes on ice, a smaller penalty for those who seldom play. Indeed, running the regression without `standardize=FALSE` leads to a bunch of farm-team players coming up on top.

Now how are we going to look at this output, with nearly 3000 coefficients? Patiently.

Start with $\hat{\alpha}$, the home-ice advantage (ignoring everything else).

```
exp(coef(nhlreg)[1])
```

```
## [1] 1.08
```

- Home ice increases the odds that the home team has scored by 8%. Without conditioning on any of the other covariates, the home team is around 8% more likely to have scored any given goal. That is a big home-ice advantage!

(De-) Selected players

Lets extract the coefficients.

```
Baicc <- coef(nhlreg)[colnames(player),]
```

By default, the reported coefficients are from the best model by **AICc**,

- a "corrected" AIC criterion.

How many are non-zero

```
c(nonzero=sum(Baicc != 0), prop=mean(Baicc != 0),
  assets=mean(Baicc > 0), liabilities=mean(Baicc < 0))
```

```
##     nonzero        prop      assets liabilities
##     646.000       0.265       0.160       0.105
```

- About 75% of the league is "average".
- 16% assets, 10% liabilities.

Top/bottom ten

Here are the top ten players. They are almost all recognizable stars.

```
Baicc[order(Baicc, decreasing=TRUE)[1:10]]
```

```
## PETER_FORSBERG   TYLER_TOFFOLI   ONDREJ_PALAT ZIGMUND_PALFFY  SIDNEY_CROSBY   JOE_THORNTON
##          0.755           0.629          0.628          0.443          0.413          0.384
##  PAVEL_DATSYUK   LOGAN_COUTURE      ERIC_FEHR MARTIN_GELINAS
##          0.376           0.368          0.368          0.358
```

And the bottom ten. They are not those with little ice time, but rather those with much ice time who underperform.

```
Baicc[order(Baicc)[1:10]]
```

```
##     TIM_TAYLOR  JOHN_MCCARTHY P. J._AXELSSON NICLAS_HAVELID    THOMAS_POCK  MATHIEU_BIRON
##         -0.864         -0.565         -0.428         -0.385         -0.384         -0.351
##  CHRIS_DINGMAN   DARROLL_POWE RAITIS_IVANANS   RYAN_HOLLWEG
##         -0.334         -0.334         -0.313         -0.299
```

Let us compare to what would happen if we run the regression without `standardize=FALSE`.

```
nhlreg.std <- gamlr(X, y, free=1:(ncol(config)+ncol(team)),
                  family="binomial")
Baicc.std <- coef(nhlreg.std)[colnames(player),]
Baicc.std[order(Baicc.std, decreasing=TRUE)[1:10]]
```

```
##        JEFF_TOMS       RYAN_KRAFT     COLE_JARRETT   TOMAS_POPPERLE    DAVID_LIFFITON
##            1.738            1.483            1.212            1.111             1.097
## ALEXEY_MARCHENKO     ERIC_SELLECK      MIKE_MURPHY       DAVID_GOVE        TOMAS_KANA
##            1.030            1.006            0.960            0.926             0.879
```

Contribution to goal for/against

Whenever a goal is scored,

- Pittsburg's odds of having scored (rather than being scored on) increase by 51% if Sidney Crosby is on the ice;

```
exp(Baicc["SIDNEY_CROSBY"])
```

```
## SIDNEY_CROSBY
##          1.51
```

- and the Blue Jackets' (or Kings', pre 2011-12) odds of having scored drop by 22% if Jack Johnson is on the ice.

```
exp(Baicc["JACK_JOHNSON"])
```

```
## JACK_JOHNSON
##        0.781
```

(Remember, the data is a little old.)

Cross-validation

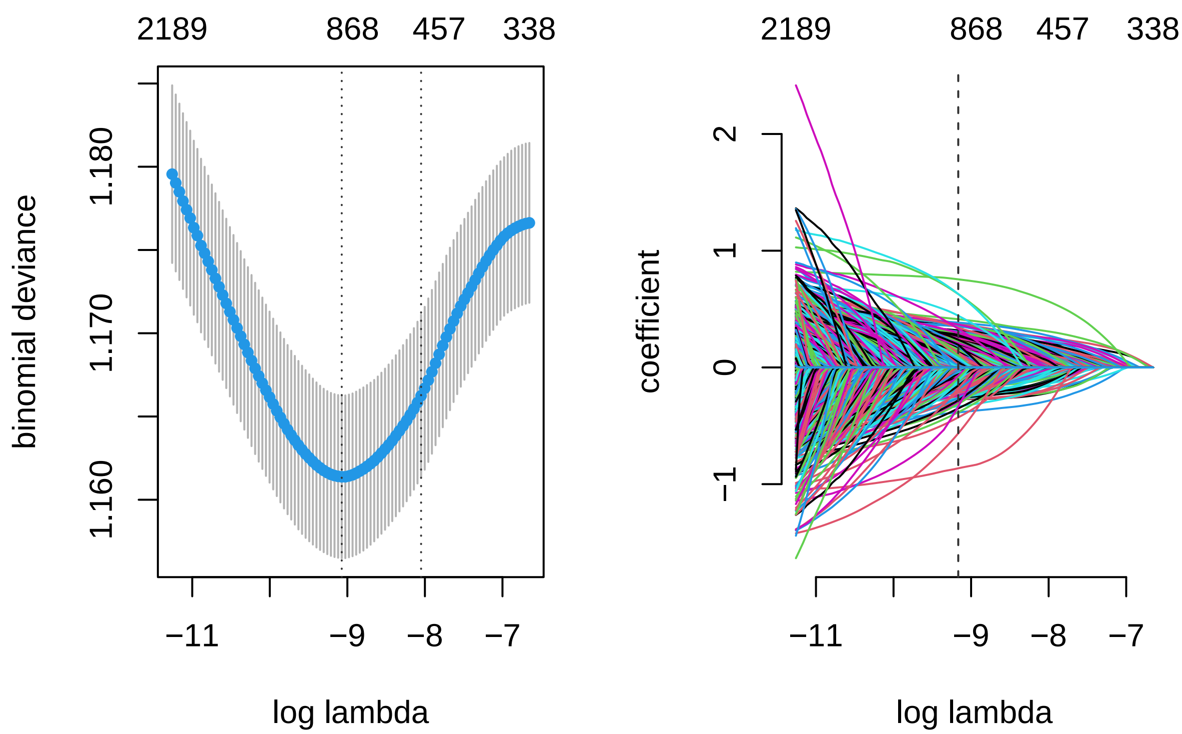
Cross-validation results instead?

```
cv.nhlreg <- cv.gamlr(X, y, free=1:(ncol(config)+ncol(team)), family="binomial",
  standardize=FALSE)
cv.nhlreg
```

```
##
## 5-fold binomial cv.gamlr object
```

The `cv.gamlr` object stores a `gamlr` object (the full data path fit) as one of its entries, and you can plot both the regularization paths and the CV experiment.

```
par(mfrow=c(1,2)); plot(cv.nhlreg); plot(cv.nhlreg$gamlr)
```



Let us look at $\log(\hat{\lambda})$ under various criteria.

```
c(AICc=as.numeric(log(nhlreg$lambda[which.min(AICc(nhlreg))])),
  AIC=as.numeric(log(nhlreg$lambda[which.min(AIC(nhlreg))])),
  BIC=as.numeric(log(nhlreg$lambda[which.min(BIC(nhlreg))])),
  CVmin=log(cv.nhlreg$lambda.min), CV1se=log(cv.nhlreg$lambda.1se))
```

```
##  AICc   AIC   BIC CVmin CV1se
## -9.17 -9.17 -6.65 -9.07 -8.05
```

Lets compare de-selection to what we got with AICc.

```
Bcvmin <- coef(cv.nhlreg, select="min")[colnames(player),]
Bcv1se <- coef(cv.nhlreg)[colnames(player),]
Bbic <- coef(nhlreg,select=which.min(BIC(nhlreg)))[colnames(player),]
c(AICc=sum(Baicc!=0), CVmin=sum(Bcvmin!=0), CV1se=sum(Bcv1se!=0), BIC=sum(Bbic!=0))
```

```
##  AICc CVmin CV1se   BIC
##   646   601   176     0
```

- Woah! BIC way over-penalizes.

Partial plus-minus

Consider the situation where you have no information beyond the fact that player "k" is on the ice.

- All other coefficients are effectively zero.

In isolation, player k's effect is the number of goals he was on the ice for, N_k, times

$$P_k - (1 - P_k) = \mathrm{P}(\text{scored}) - \mathrm{P}(\text{scored on}).$$

- I.e., his expected "goals for" in isolation is $P_k N_k$,
- and his expected "goals against" in isolation is $N_k(1 - P_k)$.

So a **partial plus-minus (PPM)** could be defined as

$$\mathrm{PPM}_k = N_k P_k - N_k(P_k - 1) = N_k(2P_k - 1).$$

- which will be on the same scale as plus-minus (PM),
- and that could help if you're not good at thinking about "log odds".

PPM calculation

Calculating PPM, and showing first 20.

```
P <- exp(Baicc)/(1+exp(Baicc))
N <- colSums(abs(player))
PPM <- N*(2*P-1)
sort(PPM, decreasing=TRUE)[1:20]
```

```
##       JOE_THORNTON      PAVEL_DATSYUK     SIDNEY_CROSBY      ALEX_OVECHKIN  HENRIK_LUNDQVIST
##                330                321               319                255               252
##       HENRIK_SEDIN       MARIAN_HOSSA  NICKLAS_LIDSTROM DANIEL_ALFREDSSON      ANDREI_MARKOV
##                237                230               224                216               213
##   MIIKKA_KIPRUSOFF     MARIAN_GABORIK   ALEXANDER_SEMIN     CHRIS_PRONGER HENRIK_ZETTERBERG
##                209                203               200                197               193
##     PETER_FORSBERG     JONATHAN_TOEWS      TEEMU_SELANNE LUBOMIR_VISNOVSKY      RYAN_GETZLAF
##                192                182               182                180               179
```

PM comparison

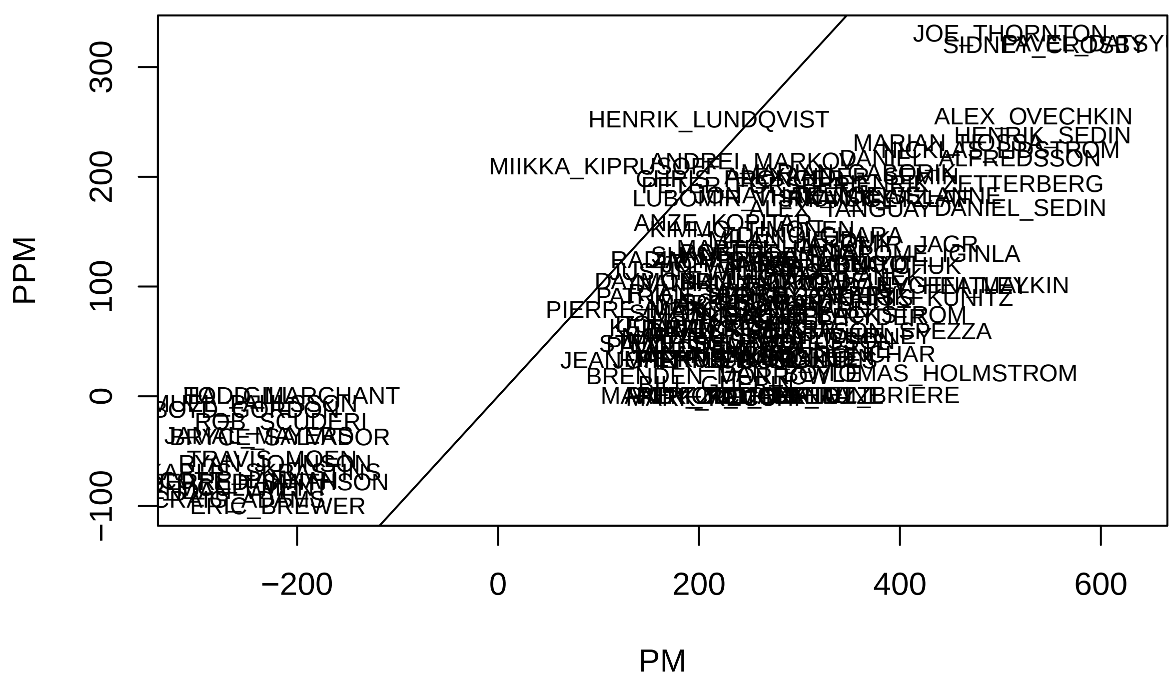
Calculating PM for comparison, and showing first 20.

- +1 for a goal by your team, −1 for a goal against.

```
PM <- colSums(player*c(-1,1)[y+1])
names(PM) <- colnames(player)
sort(PM, decreasing=TRUE)[1:20] # all goalies
```

```
##      PAVEL_DATSYUK      SIDNEY_CROSBY       HENRIK_SEDIN      ALEX_OVECHKIN       DANIEL_SEDIN
##                599                544                542                533                520
##        JOE_THORNTON  NICKLAS_LIDSTROM      EVGENI_MALKIN HENRIK_ZETTERBERG DANIEL_ALFREDSSON
##                510                500                473                471                470
##    TOMAS_HOLMSTROM       MARIAN_HOSSA       DANY_HEATLEY       CHRIS_KUNITZ      JAROME_IGINLA
##                451                448                436                426                425
##       JASON_SPEZZA      TEEMU_SELANNE       JAROMIR_JAGR       RYAN_GETZLAF      DANIEL_BRIERE
##                399                397                387                379                366
```

```
bigs <- which(abs(PM)>200|abs(PPM)>200)
plot(PM[bigs],PPM[bigs],type="n", xlim=range(PM)*1.05, xlab="PM", ylab="PPM")
text(PM[bigs],PPM[bigs],labels=colnames(player)[bigs], cex=0.75); abline(a=0,b=1)
```



If you're interested ...

If you want to read more, check out

- Original paper in the Journal of Quantitative Analysis in Sports.

    - arXiv version
    - Describes optimal line formation, and cost-benefit analysis via salary.
- Book chapter in the Handbook of Statistical Methods and Analyses in Sports.
    - arXiv version

Not sure if you'll see PPM on ESPN and time soon.