The Journal of Prediction Markets (2012) ***6 1****, 1-13*

A HIERARCHICAL BAYESIAN ANALYSIS OF HORSE RACING

Noah Silverman, MS
UCLA Department of Statistics
noahsilverman@ucla.edu

1. INTRODUCTION

Horse racing is the most popular sport in Hong Kong. Nowhere else in the world is such attention paid to the races and such large sums of money bet. It is literally a "national sport". Popular literature has many stories about computerized "betting teams" winning fortunes by using statistical analysis.[1] Additionally, numerous academic papers have been published on the subject, implementing a variety of statistical methods. The academic justification for these papers is that a parimutuel game represents a study in decisions under uncertainty, efficiency of markets, and even investor psychology. A review of the available published literature has failed to find any Bayesian approach to this modeling challenge.

This study will attempt to predict the running speed of a horse in a given race. To that effect, the coefficients of a linear model are estimated using the Bayesian method of Markov Chain Monte Carlo. Two methods of computing the sampled posterior are used and their results compared. The Gibbs method assumes that all the coefficients are normally distributed, while the Metropolis method allows for their distribution to have an unknown shape. I will calculate and compare the predictive results of several models using these Bayesian Methods.

2. OVERVIEW OF PARIMUTUEL RACING

At the racecourses in Hong Kong, the games are truly parimutuel. The betters all place their bets in a "pool" which is subsequently divided amongst the winners immediately at the end of each race. Various pools exist representing different betting combinations, but for this paper I will focus on the "win pool" which represents bets on a given horse to win the race. Unlike a casino, the track does not bet against the public but takes a fixed percentage from each betting pool. (18% in Hong Kong) The pool is divided amongst the winning betters proportionally, based upon the amount they bet.

A large tote-board at the track displays the expected winnings per dollar bet for each horse. These are commonly called the "odds", and are often mistakenly interpreted, by naive betters, as a horse's probability of winning.

What the posted odds do represent are a measure of the aggregate public opinion about a horse's likelihood to win the race. Empirical study shows that there is a 40% correlation between the public's opinion as represented by payoff odds and a horse's finishing position. The market may be considered weakly efficient.

3. LITERATURE REVIEW

Since the advent of horse racing, people have searched for a way to profit from the game. In 1986, Boltman and Chapman describe a multinomial logit model that forms the basis for most modern prediction methods.[2]. In that paper they describe a logistic regression model for predicting the "utility" of a horse. Mildly positive results (profits) were produced.

In 1994 Chapman published a second paper that refined the concepts of his first paper, while applying it to the horse racing industry in Hong Kong.[3] He compared his predicted results to the public's and found that a combination of the two produced the most profitable results.

In 1994, Bill Benter published what many consider to be the seminal work on the subject titled, "Computer Based Horse Race Handicapping and Wagering Systems"[7]. In the paper, Benter develops a two-stage prediction process. In stage one, he uses a conditional logit to calculate the "strength" of a horse. In the second stage, he combines the strength measure with the public's predicted probability using a second conditional logit function. Benter reports that his team has made significant profits during their 5 year gambling operation. (Unlike the other academics discussed here, Benter actually lived in Hong Kong and conducted a real betting operation.)

In 2007, Edelman published an extension of Benter's two-stage technique. Edelman proposes using a support vector machine instead of a conditional logit for the first stage of the process. Edelman's rationale is that a SVM will better capture the subtleties between the data. He theorizes that the betting market is near-efficient and that the bulk of the information about a horse is already contained in its Market odds for the race. His method simplifies Benters in that it only combines odds from a horse's last race with the outcome and conditions of that race and the conditions of the race today.

Lessman and Sung, in 2007 then expanded on Edelman's work by modifying the first-stage SVM process [5]. They theorized that because only jockey's in the first few finishing positions are trying their hardest; information from later finishers is not accurate as they are not riding to their full potential. The authors develop a data importance algorithm named Normalized Discounted Cumulative Gain where they assign weights to horse's data as a factor of their finishing position. The result is that data from the first place finishers is more important than the latter finishers. This NDCG is used to tune the hyperparameters of the SVM which is then subsequently used as part of the traditional two-stage model.

In 2009, Lessman and Sung published another paper expanding on their work from 2007[6]. Where previous work has focused on regression of finishing position, they chose to train an SVM based on classification [win,lose] of race results. Their argument for this approach is that it eliminates pollution of the data by potentially corrupt rank orderings, especially among minor placings. Additionally, they pre-process the data by standardizing the continuous variables *per race.*

4. DATA COLLECTION AND DESCRIPTION

Historical data from September 2005 through December 2009 were downloaded directly from the Hong Kong Jockey Club's official website. The data are 36,006 observations from 2,973 distinct horse races. (A race may have anywhere from 8 to 14 entrants.) The data was loaded into an SQL based database (MySQL). From this database of historical data, the variables were calculated and formatted into a second CSV file, using a Perl script, appropriate for direct import into R.

The single outcome variable used in this study, is the running speed of a horse expressed in meters per second. A detailed description of the covariates is included in The Code Book 2 and a sample of the data is included in Table 2.

There are two race courses in Hong Kong (Sha Tin and Happy Valley). Sha Tin has two separate tracks (turf and dirt), so there are a total of three possible tracks a race may compete on. Furthermore, horses are divided into "classes" based upon ability. The racing stewardship attempts to create a fair race by grouping horses of similar ability into the same class. Lastly, there are several distances for which races are run. All combined, there are 73 different combination of course, track, class and distance, each of which will be referred to as a *race profile.* The distribution of speed run in each profile is notably different. This may be visualized by overlaying plots of the density of speed of all race profiles onto a single plot. (Figure 1) and the boxplots of speed stratified by profile (Figure 2)

The distinctly different shape of the speed distributions suggest that a Bayesian hierarchical regression model would be well suited for this study. Following the methods of Lessman And Sung.[6], who centered their data per race, I centered the data *per profile* using the following formula:

$$\tilde{X}_{ki}^{j} = \frac{X_{ki}^{j} - \bar{X}_{k}^{j}}{\sigma_{k}^{j}} \qquad (1)$$

Where $\tilde{X}_{ki}^{j}$ (X_{ki}^{j})denotes the new (original) value of attribute t of runner i in profile j and the mean $\bar{X}$ as well as the standard deviation σ_{k}^{j} are calculated over the horses in profile j.

Speed, the dependent variable shows some correlation with the other variables, as described in Table 1. The data was divided into a *training set*, which consists of races prior to January 1st, 2009, and a *test set* consisting of races run during 2009. A Bayesian MCMC approach will be used to estimate the running speed of a horse, based on the covariates in the training set. Then model performance will be tested on the test data set.

5 THE HIERARCHICAL MODEL

The goal is to capture and describe the between-profile heterogeneity of the observations. As Hoffs[10] gives in Chapter 11, the within-profile sampling model is:

$$Y_{ij} = \beta_j^T x_{ij} + \varepsilon_{ij}, \{\varepsilon_{ij}\}: i.i.d\ \ normal(0, \sigma^2) \tag{2}$$

Where x_{ij} is a vector of variables for observation i in group j, β_j are the coefficients of the regression, and ε_{ij} are errors.

Which gives the within-group regression model of:

$$\beta: MVN(\theta,\ \Sigma) \tag{3}$$

$$Y_{i,j} = \beta_{i,j}^T x_{ij} + \varepsilon_{ij} \tag{4}$$

$$= \theta^T x_{i,j} + \gamma_j^T x_{ij} + \varepsilon_{ij} \tag{5}$$

With Θ and β are fixed and unkonw parameters to be estimated, and γ_j representing random effects that vary for each group.

5.1 Priors

An important step in Bayesian modeling is the calculation of the prior. An OLS regression was performed for each group and the resulting β_j from each group were used as the prior for the Hierarchical model.

$$\beta_j = \theta + \gamma_j \tag{6}$$

$$\gamma_1 \ldots, \gamma_j: i.i.d\ \ Multivariate\ \ Normal(0, \Sigma) \tag{7}$$

5.2 Conditional Distribution

$$Var\left[\beta_j | y_j, X_j, \sigma_j^2, \theta, \Sigma\right] = (\Sigma^{-1} + X_j^T X_j/\sigma^2)^{-1} \tag{8}$$

$$E\left[\beta_j | y_j, X_j, \sigma_j^2, \theta, \Sigma\right] = (\Sigma^{-1} + X_j^T X_j/\sigma^2)^{-1}(\Sigma^{-1}\theta + X_j^T y_j/\sigma^2) \tag{9}$$

$$\{\theta|\beta,\Sigma\}: MVN(\mu_m,\Lambda_m) \tag{10}$$

$$\Lambda_m = (\Lambda_0^{-1} + m\Sigma^{-1})^{-1} \tag{11}$$

$$\mu_m = \Lambda_m(\Lambda_0^{-1}\mu_0 + m\Sigma^{-1}\bar{\beta}) \tag{12}$$

$$\{\Sigma|\theta,\beta\}: inverse - Wishart\ (\nu_0 + m, [S_0 + S_\theta]^{-1} \tag{13}$$

$$\Sigma_\theta = \sum_{j=1}^{m}\ (\beta-\theta)(\beta-\theta)^T \tag{14}$$

6 FITTING THE MODEL WITH MCMC

6.1 Implementation of Gibbs method

First, I wrote a custom R script to simulate draws from the posterior using the Gibbs method. The initial tests showed some autocorrelation from the MCMC chains, so the code was adjusted to only sample one out of every 10 runs of the chain. A total of 300,000 iterations through the chain produced 30,000 samples from the posterior. The chain converged well after 200,000 iterations, so the final 100,000 were used as samples form the converged posterior. Storing one out of every 10 iterations gave me a chain of 10,00 draws from the converged posterior. Additionally, I calculated the Residual Sum of Squared Error (RSS) for each run and stored the results along with each posterior sample. This chain of 30,000 RSS errors allowed me to track the accuracy of the inference. The residual sum of squares for this method converged to: 988.637 which gives a predicted σ^2 of .0350 for an individual horse.

6.2 Implementation of Metropolis Hastings method

Next, I wrote custom R code to generate draws from the posterior using Metropolis Hastings. Following the same model as the Gibbs technique above, The Metropolis acceptance ratio was used as described by the formula:

$$r = \frac{p(\theta^*|y)p(\theta^*)}{p(\theta^S|y)p(\theta^S)} \tag{15}$$

Initially the acceptance ratio was low, so the variance of Σ was adjusted through trial and error to $0.5 \cdot \Sigma$ which produced an reasonable acceptance ratio of 0.54. A total of 200,000 iterations were run. The chains converged after 100,000 iterations, so the resulting100,000 were used as samples from the converged posterior. SInce I am storing 1 out of every 10, the ending posterior chain was 10,000 long. The residual sum of squares for the this

method was 1033.35 which gives a predicted σ^2 of 0.0363. This was, surprisingly, slightly higher than the RSS from the Gibbs technique.

7 RESULTS AND CONCLUSION

Predicted speeds were calculated for each horse in the test data set, to measure predictive ability of the Gibbs model. 10,000 β were drawn and then used in a regression with the covariates for each horse. The maximum a-posteriori (MAP) of the resulting regression for each horse was stored as "predicted speed". The variance of this predicted speed was 0.0397. The horse with the fastest predicted speed won his race 21.63% of the time. This is better than a random choice, which would produce an expected winner between 7.14% and 12.5% (Depending on the number of horses in a race). However, simply betting on the horse with the highest predicted speed was not enough to profit. (The ultimate goal is not to guess winners, but generate profit.)

As a further step, a conditional logit as calculated to combine our predicted speed with the public's odds estimate. As this is the "standard" for other predictive models. The coefficients for that model were 6.4186 for the public odds and 4.0541 for the hierarchical Bayesian predicted speed.

As a further test of performance, the expected value for each bet was calculated as $ev = payoff\ if\ won\ x\ probability\ of\ winning$. There were 3,323 horses in the test set with positive expected value (out of 8,618 possible.) If $1 had been bet on each horse with a positive expected value, the return would have been 2919 resulting in a net loss of $404 (-12.15%). While the predictive errors are small, the model is still not good enough to generate profit without further refinement.

8 ACKNOWLEDGEMENTS

The author would like to thank Juana Sanchez, Senior Lecturer, at UCLA for all her help, and for teaching the course C236 Bayesian Statistics, where he conceptualized the ideas for this paper.

REFERENCES

[1] Michael Kaplan, *The High Tech Trifecta*. Wired Magazine, October 2003

[2] R.N. Bolton and R.G. Chapman, *A multinomial Logit Model For Handicapping Horse Races*. Efficiency of Racetrack Betting Markets, Academic Press, Inc. 1994

[3] Randall G. Chapman, *Still Searching For Positive Returns At the Track: Empirical Results From 2,000 Hong Kong Races* Efficiency of Racetrack Betting Markets, Academic Press, Inc. 1994

[4] David Edelman, *Adapting support vector machine methods for horserace odds prediction*. Ann Oper Res (2007) 151:325-336, Springer Science + Business Media
[5] Stefan Lessman, Ming-Chien Sung, and Johnnie E.V. Johnson, *Adapting Least-Square Support Vector Regression Models to Forecast the Outcome Of Horseraces* The Journal of Prediction Markets (2007) 1 3, 169-187
[6] Stefan Lessman, Ming-Chien Sung, and Johnnie E.V. Johnson, *Identifying winners of competitive events: A SVM-based Classification Model for Horserace Prediction* European Journal of Operational Research 196 (2009) 569-577
[7] Bill Benter, *Computer Based Horse Race Handicapping and Wagering Systems: A Report*. Efficiency of Racetrack Betting Markets, Academic Press, Inc. 1994
[8] Yulanda Chung, *A Punter's Program Makes Millions Trackside* AsiaWeek Magazine, November 3, 2000 Vol. 26 No. 43
[9] Michael Kaplan, *Gambling: The Hundred and Fifty Million Dollar Man*, Cigar Aficionado Magazine
[10] Peter D. Hoff *A First Course in Bayesian Statistical Methods*, Springer Science and Business Media 2009

APPENDIX

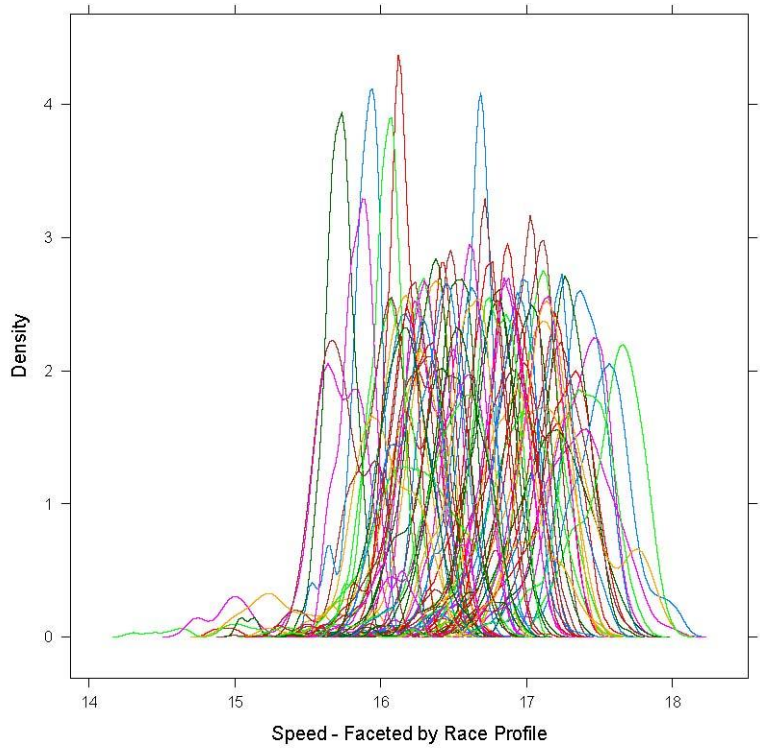

Figure 1: Density distributions of speed stratified by profile

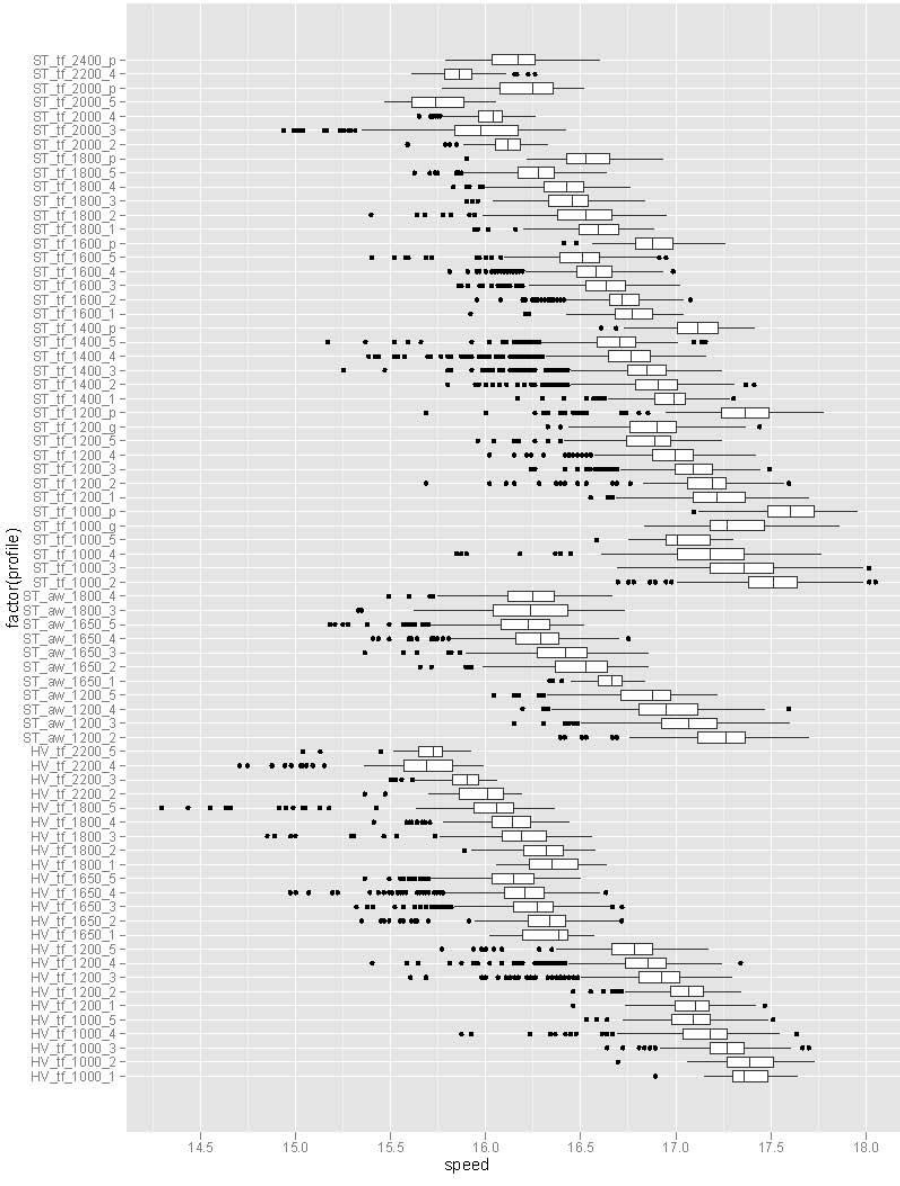

Figure 2: Boxplots of speed stratified by race profile

Table 1. Correlation of variables to speed

Name	Correlation
last rank	0.077898638
last run 1	0.091421415
last run 2	0.106251733
last run 3	0.110888381
last odds prob	0.111767807
last distance	-0.562025513
last weight	0.035323525
last draw	0.007378011
last speed	0.563767210
last percentage	-0.016219487
perc won	0.159669358
last 4 perc	-0.024847700
last 4 rank	0.128202630
last 4 odds prob	0.147430995
rest	0.136668197
distance	-0.799783155
weight	0.005454088
draw	0.022529607
total races	-0.187955730
bad runs	-0.077079847
jockey rides	-0.030171224
dist last 30	-0.217824843

Table 2: Data Code Book

Name	Mean	SD	Low	High	Notes
speed	16.648026	0.421332	14.297061	18.050542	Speed run in the race. This is our dependent variable
odds prob	0.082121	0.081307	0.002158	0.745455	Public estimated probability of winning
last rank	0.000457	1.000038	-1.672634	1.577826	Finishing position in last race
last run 1	0.000310	0.999845	-1.613067	1.594531	Position at first quarter of last race
last run 2	0.000365	0.999841	-1.623586	1.597088	Position at second quarter of last race
last run 3	0.000191	0.999820	-1.657603	1.597270	Position at third quarter of last race
last odds prob	0.000544	0.999850	-0.976207	8.181944	Public estimated probability of winning last race
last distance	-0.000505	0.999136	-1.679878	3.419010	distance of last race
last weight	-0.000349	1.000267	-2.866383	1.830325	weight carried in last race
last draw	0.000408	1.000169	-1.566456	1.879832	post position of last race
last speed	0.000237	0.999575	-5.522472	3.813677	speed run in last race
last percentage	0.000407	0.999548	-7.595383	2.543021	speed as percentage of profile record in last race
perc won	-0.000762	0.996582	-0.766225	7.839123	percentage of races won in lifetime
last 4 perc	0.001131	0.997601	-7.240364	2.922523	mean speed as percentage of profile record in last 4 races
last 4 rank	0.000107	0.999540	-2.348644	2.335103	mean finishing position in last 4 races
last 4 odds prob	0.000203	0.999670	-1.132926	9.931562	mean public estimated probability in last 4 races
rest	0.000100	1.000661	-0.991545	5.016491	days rest since last race
distance	-0.002648	0.996482	-1.726967	3.412664	distance of this race
weight	0.000140	0.999623	-2.827238	1.807332	weight carried in this race
draw	0.000548	1.000079	-1.568536	1.882718	post position in this race
total races	0.001505	1.000234	-1.321604	4.875548	total races run during lifetime
bad runs	0.000264	1.000246	-0.293411	8.160743	failure to finish in lifetime
jockey rides	0.000219	0.999542	-0.651017	10.310366	number of times jockey has ridden this horse
dist last 30	0.000271	0.999912	-1.167798	6.151917	distance run in the last 30 days

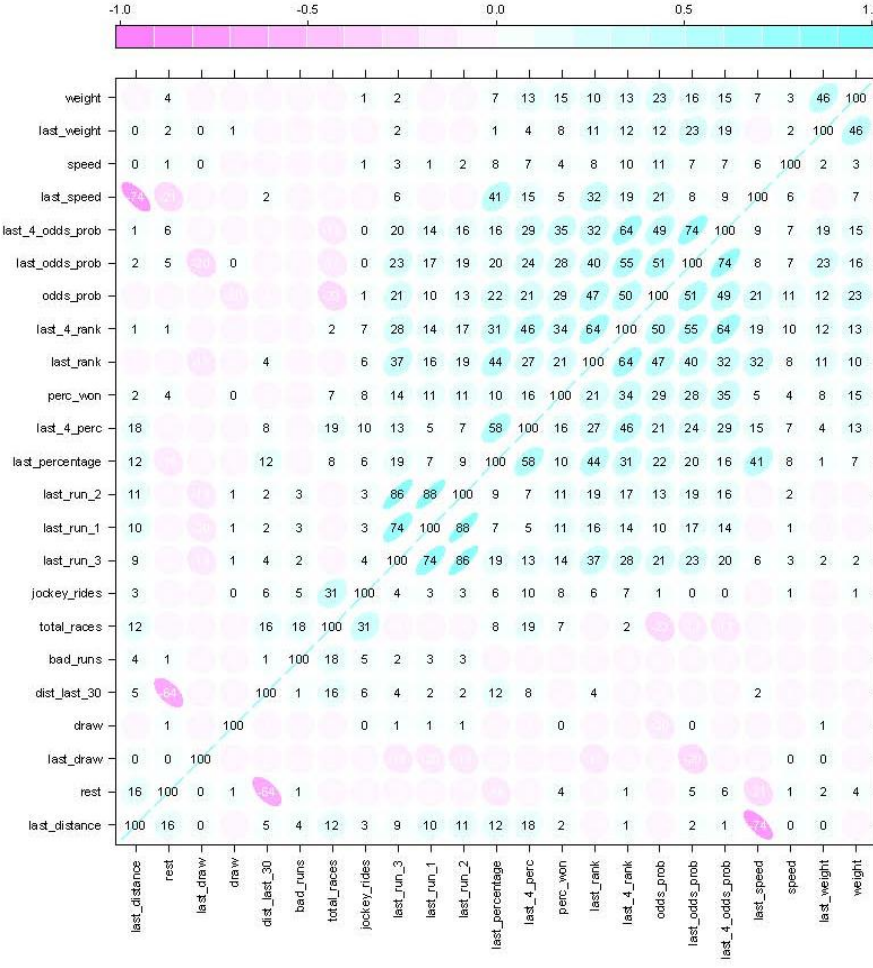

Figure 3. Graphical representation of correlation of variables

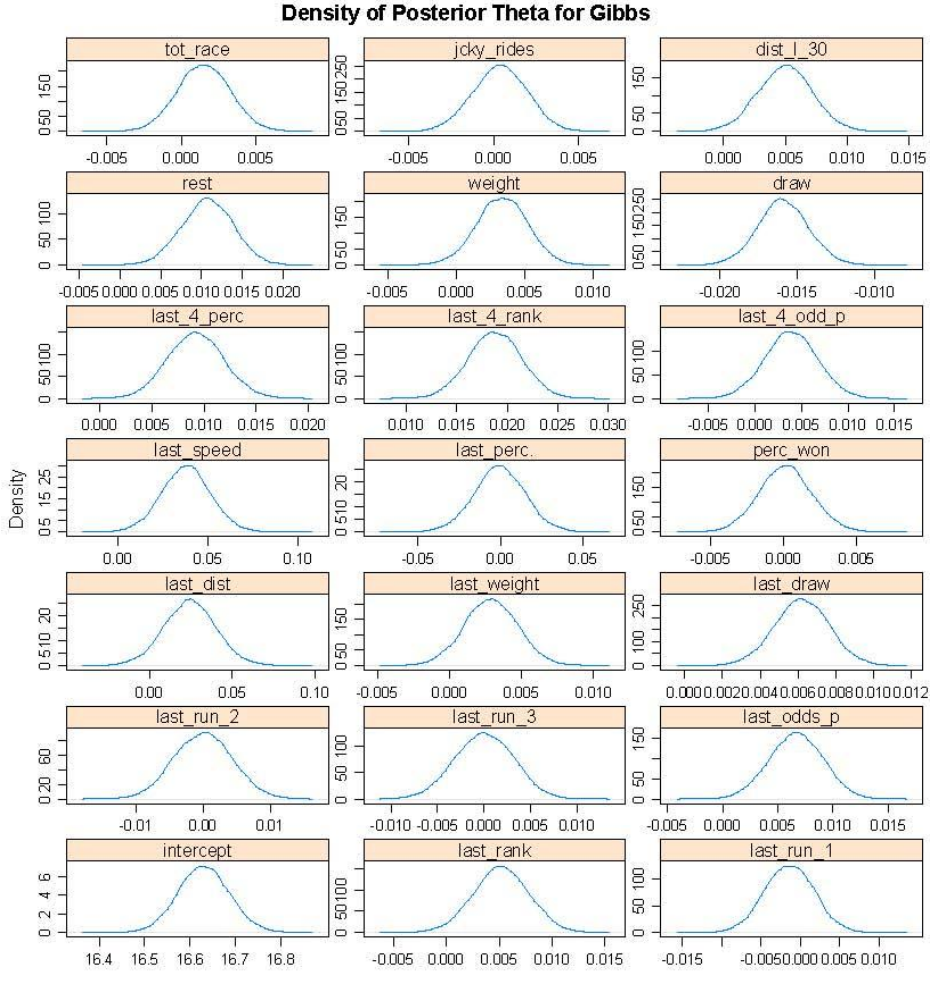

Figure 4. Posterior density of theta for predictor variables –Gibbs

Table 3: Sample of data

	1	2	3	4	5	6	7	8	9	10
1	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
last_rank	-0.98	-1.34	1.48	-1.29	0.56	1.53	1.53	0.75	-0.30	-0.37
last_run_1	0.15	-0.36	-1.60	-0.72	1.32	0.87	1.12	-0.36	0.13	1.61
last_run_2	-1.30	0.12	-1.30	-1.59	1.58	-1.59	1.10	-0.61	-1.10	1.01
last_run_3	-1.37	0.36	0.13	-1.37	1.62	-0.91	1.37	-0.40	-0.66	1.32
last_odds_prob	0.40	0.16	0.63	-0.39	-0.78	-0.62	-0.28	-0.14	-0.91	0.16
last_distance	-1.12	-1.12	-1.12	1.06	1.06	0.82	-0.15	-1.12	-1.12	1.06
last_weight	-0.15	1.36	-0.83	0.40	-0.56	0.81	0.67	-0.83	-1.11	0.54
last_draw	0.95	-1.12	-1.12	0.95	-0.86	1.73	-0.08	-1.38	0.18	0.18
last_speed	0.10	0.53	0.40	-1.30	-1.09	-0.08	0.65	0.04	-0.44	-1.75
last_percentage	-0.43	-1.00	-0.88	0.40	0.76	0.14	0.72	-1.79	-2.57	-0.38
perc_won	-0.90	-0.90	-0.90	-0.90	-0.90	-0.90	-0.90	-0.90	-0.90	0.94
last_4_perc	0.55	-0.96	-0.00	-0.37	-1.28	-0.73	-0.29	-1.69	-2.03	0.08
last_4_rank	0.00	-0.32	1.25	-0.33	-0.50	0.96	1.31	1.16	-1.24	0.20
last_4_odds_prob	0.77	0.42	0.50	0.65	-0.35	0.21	-0.24	-0.34	-1.07	0.27
rest	0.72	1.52	1.00	2.53	0.90	1.16	1.16	0.79	0.79	3.80
weight	-0.20	0.26	1.03	-0.65	0.26	0.88	0.72	0.88	-0.20	-0.65
draw	-0.33	1.42	0.67	0.17	0.92	-1.34	1.67	-1.59	0.42	-1.08
total_races	-0.24	-0.30	-0.24	-0.54	-0.97	-0.24	-0.67	-1.10	-1.16	-0.48
jockey_rides	-0.19	-0.59	-0.19	-0.59	-0.39	-0.39	-0.59	-0.59	0.21	-0.59
dist_last_30	-1.06	-1.06	-1.06	-1.06	-1.06	-1.06	-1.06	-1.06	-1.06	-1.06