

I. Introduction

Nike recently released a running shoe, called the Vaporfly, that has come under great scrutiny due to its potential performance boosting effects. The objective of our study is to determine if these claims are valid by estimating the effect on marathon times in wearing the Vaporfly shoes. Three types of Bayesian linear models will be tested for accuracy using JAGs to determine the best model. If there is an improvement, we will also test whether it varies across gender, runner, and/or the course. I will use the men's marathon running data to find and use the best model to determine the improvement across runner and/or course. From there, I will fit the model for the womens marathon runners and see if there is a significant difference in improvement across genders.

II. Methods

The models I propose will be three different types of Bayesian linear models fit with JAGS. The first will be a varying intercept and slope random effects, the second a mixed effects (varying slope, intercept fixed), and the third a fixed effects model. The random effects in this model are runner effects and marathon location effects. The models are displayed below.

- 1) A varying slopes and intercepts model, where random effects will adjust both the fixed intercept and slope. Suppose the Marathon Time is modeled as Y_{ijk} with random effects R and M , for runner and marathon random effects. The condition of if the runner is wearing Vaporfly shoes is represented by V . Then the full model is expressed as this sum.

$$Y_{ijk} = \beta_0 + R_{0j} + M_{0k} + (\beta_1 + R_{1j} + M_{1k}) * V_i + \varepsilon_{ijk} \quad (1)$$

- 2) A varying slopes and fixed intercept model, where random effects will adjust only the slope. That is, all subjects will have some fixed mean in this case.

$$Y_{ijk} = \beta_0 + (\beta_1 + R_{1j} + M_{1k}) * V_i + \varepsilon_{ijk} \quad (2)$$

- 3) A fixed effects model which is a simple linear regression with the Vaporfly condition as the sole predictor.

$$Y_i = \beta_0 + (\beta_1) * V_i + \varepsilon_i \quad (3)$$

The random effects are normally distributed with mean zero and an unknown variance. Model 1 also assumes that the intercepts and slopes for marathon runners and intercepts and slopes for location are correlated. We can obtain a covariance matrix of a random effect by Equation 4 below.

$$\Sigma_M = \begin{pmatrix} \sigma_{M_0}^2 & Cov(\sigma_{M_0} \sigma_{M_1}) \\ Cov(\sigma_{M_0} \sigma_{M_1}) & \sigma_{M_1}^2 \end{pmatrix} \quad (4)$$

$$Cov(\sigma_{M_0} * \sigma_{M_1}) = \rho_M \sigma_{M_0} \sigma_{M_1} \quad (5)$$

By finding the variance of the random effects by runner and location for each respective slope and intercept, we can theoretically measure exactly how they affect the magnitude of the conditional Vaporfly effect.

There were $M = 22$ different marathons and $R = 308$ unique racers as factors for the random effects. For Model 1, random effects were modeled as multivariate normal since there was a slope and intercept component. Fixed effects were normal, with the slope's mean set at zero and the intercept's mean at 140 minutes – a number close to the average men's race. Random effects had means of zero for intercept and slope, and Equations 4 and 5 were used to model the covariance matrix for the random effect. A Wishart distribution using this matrix gave the estimate for the precision. The inverse of this matrix gave the covariance matrix for the random effect. This process was repeated for the other random effect.

Model 2 was simpler since only the slope was varying, the intercept was fixed. Priors for the fixed effects were the same as in Model 1. The random effects only varied with slope, with mean zero and variance set as a gamma distribution. The residual variance was also a gamma distribution prior.

Model 3 was the simplest. The fixed priors were the same and the residual variance was a gamma distribution prior.

III. Computation

The models were run with MCMC sampling using 'rjags' in R. There was a burn-in period of 10,000 iterations and 40,000 total iterations were sampled. The thinning interval was 20, number of chains was 2, and the sample size per chain was 1,500. The trace plots indicated convergence for each model. Model 1's trace plots for the fixed parameters and residual variance are shown below.

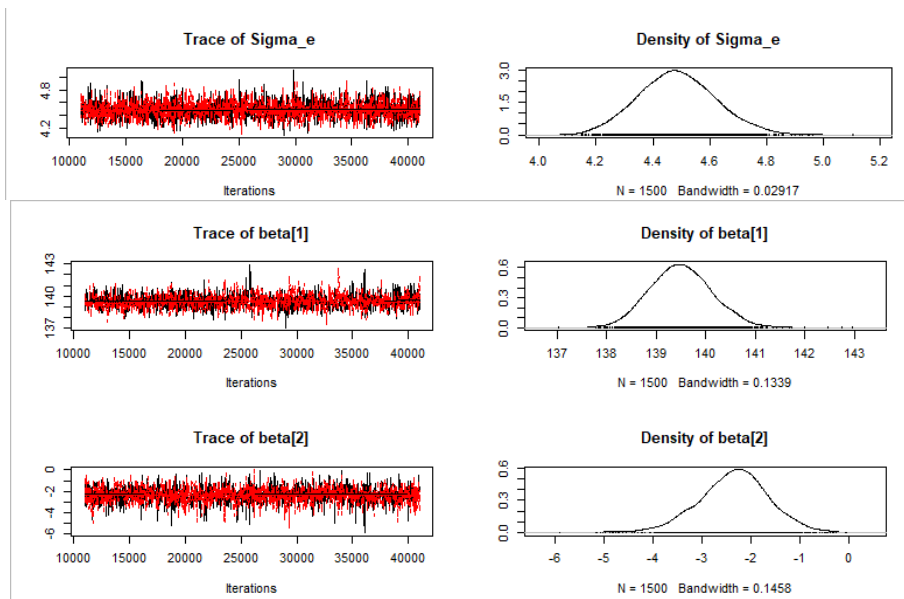


Figure 1: Trace Plots for Model 1

IV. Model Comparisons

Below is a table of DIC model selection. The goal is to minimize this number, as the smallest number DIC indicates the best model with high probability.

Table 1: Model Comparison using DIC

Model Number	Mean Deviance	Penalty	Total Deviance
1	4887	212.6	5100
2	5519	86.44	5606
3	5544	2.995	5547

Therefore Model 1 is the best model of the three.

V. Results

Below is a table with 95% credible intervals of our parameters, residual error variance, and random effects matrix variance.

Table 2: Final Results of Model 1

	mean	sd	2.5%	50%	97.5%	Rhat	n.eff
Sigma.u[1,1]	16.96	2.09	13.20	16.84	21.40	1.00	11495
Sigma.u[2,1]	-1.05	1.34	-3.76	-0.97	1.67	1.29	20
Sigma.u[1,2]	-1.05	1.34	-3.76	-0.97	1.67	1.29	20
Sigma.u[2,2]	0.19	0.29	0.00	0.09	0.78	1.48	68
Sigma.w[1,1]	5.20	2.87	1.62	4.56	12.40	1.00	8097
Sigma.w[2,1]	-0.27	1.79	-4.62	-0.06	2.83	1.00	3123
Sigma.w[1,2]	-0.27	1.79	-4.62	-0.06	2.83	1.00	3123
Sigma.w[2,2]	1.59	2.13	0.03	0.86	7.33	1.00	2255
Sigma_e	4.48	0.14	4.22	4.48	4.76	1.00	18360
beta[1]	139.51	0.64	138.30	139.49	140.82	1.00	1988
beta[2]	-2.31	0.68	-3.74	-2.29	-1.04	1.00	6851
rho.u	0.1117						
rho.w	0.2059						

By looking at beta[2], we see that the 95% credible set for men wearing Vaporfly is a reduction of (1.04, 3.74) minutes. This interval does not contain zero so **it is a significant reduction in time** with this dataset. The Sigma.u values are results for the variance of the runners, and the Sigma.w values are results for the variance of the marathon locations. Using the Equation 4 and 5 to solve for variance, we get a larger variance for Sigma.u, suggesting it varied more by runners than course location.

Table 3: Final Results of Model 1 (Women's Data)

	mean	sd	2.5%	50%	97.5%	Rhat	n.eff
beta[1]	159.84	0.95	158.03	159.83	161.77	1.00	1531
beta[2]	-1.81	0.91	-3.63	-1.79	-0.07	1.00	7269

For women, the Vaporfly effect was a reduction of (.07, 3.63) minutes. This is a significant reduction for women overall, however we cannot say that it is significantly different from the men's reduction since the intervals overlap.

JAGS code

```
#model Varying intercepts and slopes

data    <- list(Y=Y,n=840, v=v, J=J, M=M,
               m_id=m_id, n_id=n_id, zero.u=zero.u)

burn    <- 10000

n.iter  <- 30000

thin    <- 20

n.chains <- 2

model_string <- textConnection("model{

# Likelihood

for(i in 1:n){

  mu[i] <- (beta[1]+w[m_id[i],1]+u[n_id[i],1])
  +(beta[2]+w[m_id[i],2]+u[n_id[i],2])*v[i]

  Y[i] ~ dnorm(mu[i], tau.e)
}

# Random effects

for(k in 1:M){

  w[k,1:2] ~ dmnorm(zero.u,invSig.w)
}

for(j in 1:J){

  u[j,1:2] ~ dmnorm(zero.u,invSig.u)
}

# priors

beta[1] ~ dnorm(140,0.001)
beta[2] ~ dnorm(0,0.001)

#error variance

Sigma_e ~ dgamma(20,1)
tau.e <- pow(Sigma_e,-2)

#for individual random effects

tau.u1 ~ dgamma(1,.01)
tau.u2 ~ dgamma(1,.01)
```

```

sigma.u1 <- pow(tau.u1,-1/2)
sigma.u2 <- pow(tau.u2,-1/2)

invSig.u ~ dwish(R.u,2.1)
Sigma.u <- inverse(invSig.u)
R.u[1,1] <- pow(sigma.u1,2)
R.u[2,2] <- pow(sigma.u2,2)
R.u[1,2] <- rho.u*sigma.u1*sigma.u2
R.u[2,1] <- rho.u*sigma.u1*sigma.u2

rho.u ~ dnorm(mu_rho.u,tau_rho.u)T(-1,1)
mu_rho.u ~ dunif(-1,1)
tau_rho.u ~ dgamma(.1,.0001)

#for marathon location random effects
tau.w1 ~ dgamma(.1,.1)
tau.w2 ~ dgamma(.1,.1)
sigma.w1 <- pow(tau.w1,-1/2)
sigma.w2 <- pow(tau.w2,-1/2)

invSig.w ~ dwish(R.w,2.1)
Sigma.w <- inverse(invSig.w)
R.w[1,1] <- pow(sigma.w1,2)
R.w[2,2] <- pow(sigma.w2,2)
R.w[1,2] <- rho.w*sigma.w1*sigma.u2
R.w[2,1] <- rho.w*sigma.w1*sigma.u2

rho.w ~ dnorm(mu_rho.w,tau_rho.w)T(-1,1)
mu_rho.w ~ dunif(-1,1)
tau_rho.w ~ dgamma(1,.0001)

})

params <- c("beta", "Sigma.u", "Sigma.w", "Sigma_e", "rho.w", "rho.u")
model <- jags.model(model_string,data = data, n.chains=n.chains,quiet=TRUE)
update(model, burn, progress.bar="none")
samples1 <- coda.samples(model, variable.names=params,
                          n.iter=n.iter, thin=thin, progress.bar="none")

```

```
plot(samples1)
```

```
summary(samples1)
```