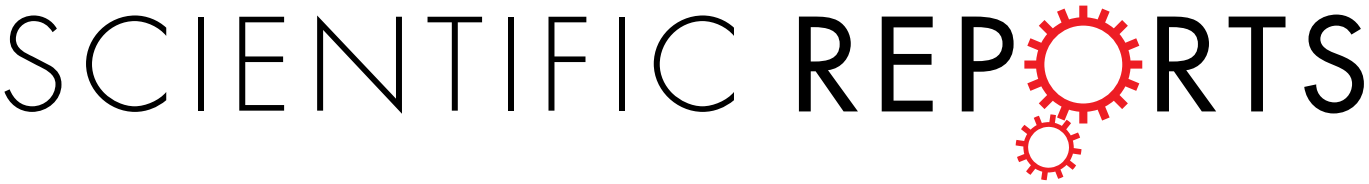

OPEN

Prediction of genomic breeding values using new computing strategies for the implementation of MixP

Received: 22 December 2016
Accepted: 25 October 2017
Published online: 08 December 2017

Linsong Dong[1], Ming Fang[1] & Zhiyong Wang[1,2]

MixP is an implementation that uses the Pareto principle to perform genomic prediction. This study was designed to develop two new computing strategies: one strategy for nonMCMC-based MixP (FMixP), and the other one for MCMC-based MixP (MMixP). The difference is that MMixP can estimate variances of SNP effects and the probability that a SNP has a large variance, but FMixP cannot. Simulated data from an international workshop and real data on large yellow croaker were used as the materials for the study. Four Bayesian methods, BayesA, BayesCπ, MMixP and FMixP, were used to compare the predictive results. The results show that BayesCπ, MMixP and FMixP perform better than BayesA for the simulated data, but all methods have very similar predictive abilities for the large yellow croaker. However, FMixP is computationally significantly faster than the MCMC-based methods. Our research may have a potential for the future applications in genomic prediction.

The advent of next generation sequencing technology has accelerated the development of the theory behind quantitative molecular genetics approaches, such as quantitative trait loci (QTL) mapping, genome-wide association (GWA) studies and genomic selection. Genomic selection was first proposed by Meuwissen *et al.* as an efficient method to predict animal breeding outcomes[1]. Recently, various implementations have been proposed for genomic prediction, such as genomic best linear unbiased prediction (GBLUP)[2], ridge-regression BLUP (RRBLUP), BayesA, BayesB[1], BayesCπ[3], BayesLASSO[4,5], BayesSSVS[6], fast Bayesian methods[7,8], MixP[9], among others. GBLUP and RRBLUP assume a constant variance for all SNP loci, which may be an imprecise assumption if a trait is affected by a small number of QTL loci[1]. Bayesian methods propose more flexible prior assumptions for SNP effects (or variances). Generally, the prior distributions of Bayesian methods assume that there are large variances in some SNP loci and small or even zero variances at other loci, which seems to be more realistic. The implementations of Bayesian methods, such as BayesA, B, Cπ and LASSO, are mainly based on Markov chain Monte Carlo (MCMC) algorithms, requiring much more computation time to estimate SNP effects. To increase the computational speed, researchers have suggested some fast Bayesian methods, such as fast BayesB[7] and emBayesB[8]. Yu and Meuwissen proposed another fast Bayesian method using the Pareto principle to perform genomic prediction[9].

The Pareto principle was proposed by the economist Vilfredo Pareto at the beginning of the 20th century[10]. This principle states that approximately 20% of the population possesses 80% of the wealth in a country. Similar theories have been further applied in various fields, such as in genomic prediction by Yu and Meuwissen[9], resulting in the method termed MixP. The prior distribution of MixP is a mixture of two normal distributions, which assumes that x% of the SNPs cause $(100-x)$% of the genetic variance, so the remaining $(100-x)$% of SNPs decide the remaining x% of genetic variance. Here we assume $\gamma = x$%, and $(1-\gamma) = (100-x)$%. The large and small variances are proposed as follows[9]:

[1]Key Laboratory of Healthy Mariculture for the East China Sea, Ministry of Agriculture; Fisheries College, Jimei University, Xiamen, Fujian, P.R. China. [2]Laboratory for Marine Fisheries Science and Food Production Processes, Qingdao National Laboratory for Marine Science and Technology, Qingdao, 266235, P.R. China. Correspondence and requests for materials should be addressed to Z.W. (email: zywang@jmu.edu.cn)

	$r_{(TBV,GEBV)}$	$b_{(TBV,GEBV)}$
BayesA	0.807	0.850
BayesCπ	0.885 ([a]$\pi = 0.0096$)	0.896
[b]FMixP	0.882 ($\gamma = 0.07$)	0.980 ($\gamma = 0.07$)
FMixP ($\gamma = 0.5$)	0.753	0.851
MMixP	0.885 ([c]$\gamma = 0.0092$)	0.893

Table 1. Correlation and regression coefficients of TBV on GEBV for various methods in simulated data. [a]π is the probability of a SNP with non-zero effect estimated by BayesCπ. [b]The optimized result estimated by FMixP when γ equals the value in the parentheses. [c]γ is the probability of a SNP with large variance estilmated by MMixP. $r_{(TBV,GEBV)}$ and $b_{(TBV,GEBV)}$ represent the correlation and regression coefficients of TBV on GEBV, respectively.

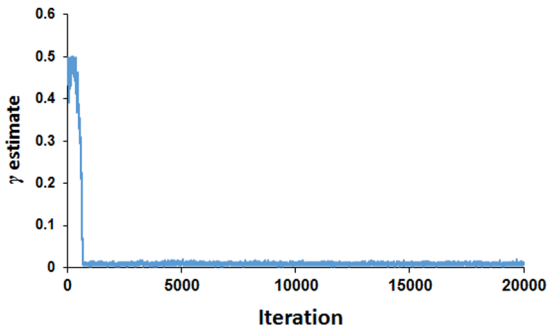

Figure 1. γ estimates in the Gibbs sampling cycles of MMixP in simulated data.

$$\begin{cases} \sigma_1^2 = \dfrac{(1-\gamma)V_g}{\gamma M} \\ \sigma_2^2 = \dfrac{\gamma V_g}{(1-\gamma)M} \end{cases}, \quad (1)$$

where σ_1^2 and σ_2^2 represent the large and small variance of a SNP effect, respectively; V_g is the total additive genetic variance; M is the number of SNPs; and $\gamma \leq 0.5$. The prior for MixP assumes that all SNPs have effects, but each SNP has only two possible variances: σ_1^2 or σ_2^2. This is similar but not completely identical to the assumptions found in two other Bayesian methods (BayesA and BayesCπ). In BayesA, the prior also assumes that all SNPs have effects but each SNP has its own variance. The variances of the SNP effects in BayesA follow an inverse-chi-squared distribution[1,11]. The prior for BayesCπ assumes that SNPs with non-zero effects have a common variance, which is similar to the assumption of MixP, which assumes that "large" SNPs have a common variance (σ_1^2). However, SNP effects with small variance may be shrunk to zero in BayesCπ.

MixP is also a fast Bayesian method that is not based on a MCMC algorithm[9]. However, a multivariate normal density and an inverse matrix are included in the derivation, increasing the difficulty in understanding the derivation. In the nonMCMC-based MixP, the γ is given but not estimated, such that the optimal value of γ should be searched using a cross-validation. However, the parameter γ can be estimated using the MCMC algorithm. For the sake of convenience in distinguishing different algorithms, the MixP not based on the MCMC algorithm is termed fast MixP (FMixP), and the MixP based on the MCMC algorithm is termed MCMC-based MixP (MMixP) here.

In this study, we developed two new computing strategies for FMixP and MMixP, respectively. The first strategy used a product of univariate densities instead of the multivariate normal density to estimate SNP effects for FMixP; the second strategy attempted to use the MCMC algorithm to derive the solutions for MMixP. In addition, the strategies were used to analyse the results on simulated data from an international workshop and real data on large yellow croaker, and compared the predictive abilities with estimations by BayesA and BayesCπ.

Results

Results for simulated data. The predictive results of various Bayesian methods for the simulated data are shown in Table 1. The predictive accuracies are very close in BayesCπ, MMixP, and FMixP ($\gamma = 0.07$). The accuracy of BayesA is lower than that of BayesCπ, MMixP, and FMixP ($\gamma = 0.07$), but higher than FMixP when $\gamma = 0.5$. BayesCπ and MMixP yield comparatively accurate estimates for π and γ, respectively. As there are 48 QTLs simulated in the genome, the true value of π (or γ) is $48/5726 \approx 0.0084$, which is very close to the values estimated by BayesCπ and MMixP. The γ estimates in the Gibbs sampling cycles are shown in Fig. 1. We can find that the value converges when the Gibbs sampling runs at ~1000th cycle.

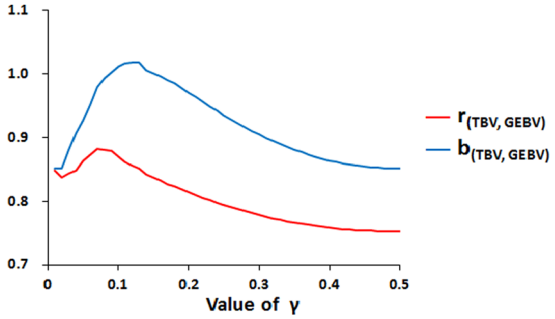

Figure 2. Graphs of the correlation and regression coefficients of TBV on GEBV for FMixP against γ in simulated data.

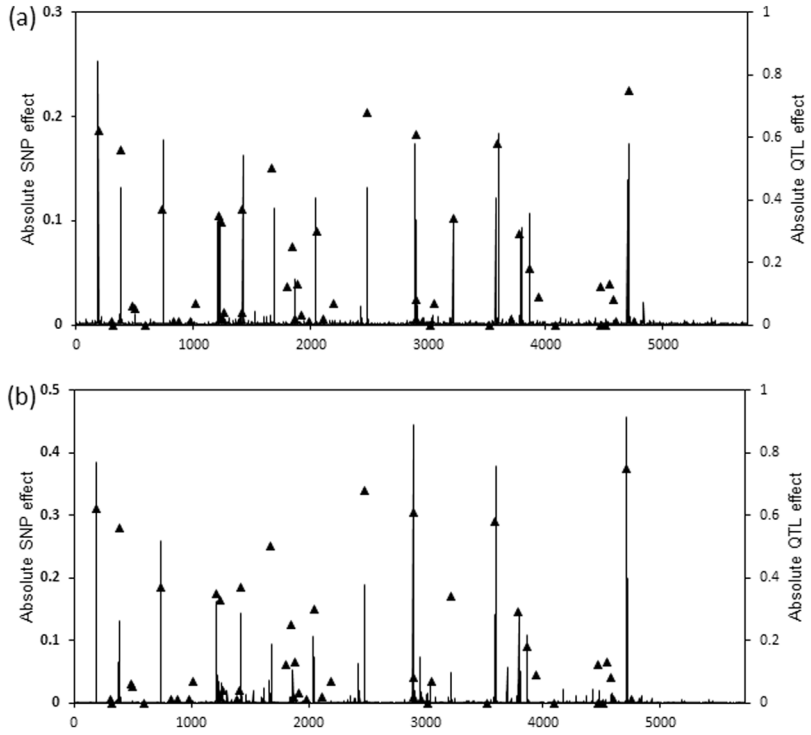

Figure 3. Distributions of absolute SNP effects estimated by FMixP and MMixP in simulated data. (**a**) FMixP with $\gamma = 0.07$; (**b**) MMixP. ▲ represents the location and effect of QTL in genome.

We compared the predictive results between MixP introduced by Yu and Meuwissen[9] and our FMixP, and found that the two derivations could yield the same prediction accuracies. Graphs of the correlation and regression coefficients of TBV on GEBV ($r_{(TBV,GEBV)}$ and $b_{(TBV,GEBV)}$, respectively) against γ for FMixP are presented in Fig. 2. Both measures of accuracy follow a similar trend in response to γ. Overall, FMixP yields the highest accuracy when the value of γ is close to 0.07, but this value is higher than the true value (0.0084). The distributions of SNP effects estimated by FMixP and MMixP are shown in Fig. 3. All the QTLs with absolute effects >0.2 can be located by the nearby SNPs in both methods, indicating that the MixP may be a promising implementation in GWA study.

Results for real data. Table 2 shows the predictive abilities of various Bayesian methods for four quantitative traits in large yellow croaker. The results estimated by BayesA, BayesCπ, MMixP and FMixP are very similar for all traits, with no within-trait difference in predictive ability greater than 0.01. The value of γ (the probability of a SNP with a large variance) estimated by MMixP is much higher than that estimated in the simulated data, indicating that there may be many QTLs affecting the phenotypes. The results of FMixP show that predictive abilities are optimized when the probability of a SNP with a large variance in specific traits is 0.02 or 0.05. However, these optimal points are not obvious because the predictive abilities are still very close to the best results when $\gamma = 0.5$, which is not consistent with the results from the simulated data. Figure 4 shows graphs of the predictive ability against γ for FMixP for various traits. It shows that the value of γ barely affects the predictive ability as long

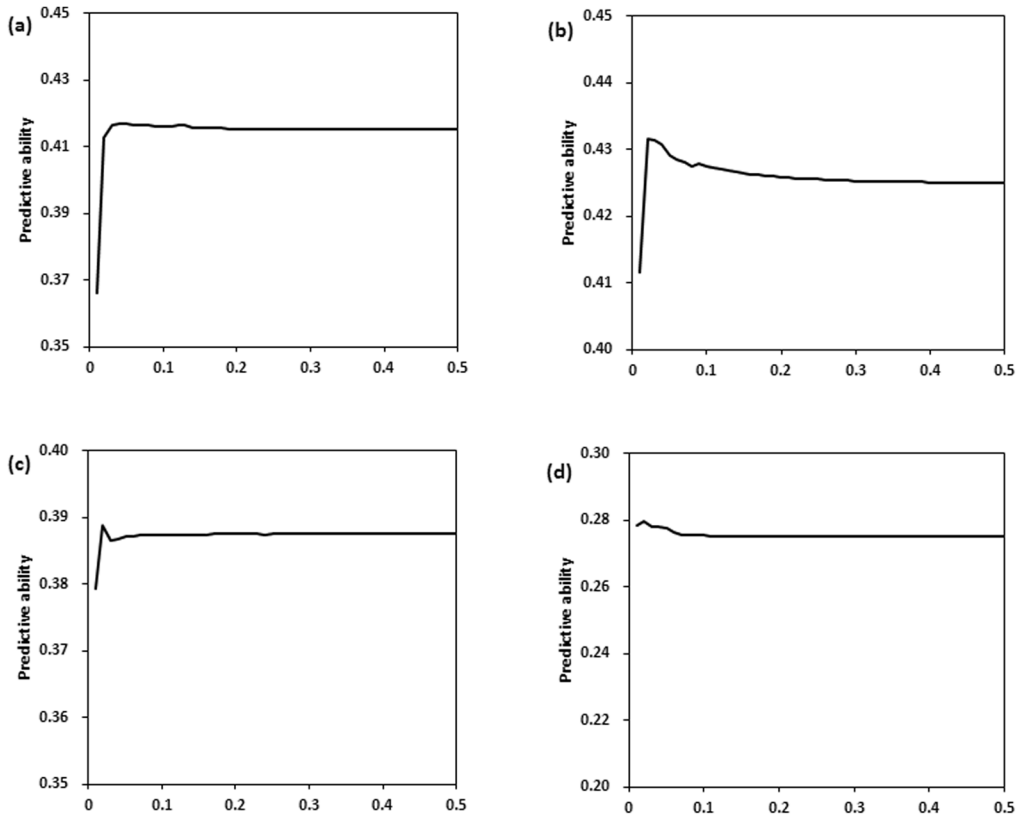

Figure 4. Graphs of the predictive ability of FMixP against γ for four traits in large yellow croaker. (**a**) Body weight; (**b**) Body length; (**c**) Body height; (**d**) Length/height.

Trait	Predictive ability (Mean ± SE)				
	BayesA	BayesCπ	[a]FMixP	FMixP ($\gamma = 0.5$)	MMixP
Body weight	0.413 ± 0.040	0.413 ± 0.040	0.417 ± 0.041 ($\gamma = 0.05$)	0.415 ± 0.040	0.412 ± 0.040
Body length	0.431 ± 0.033	0.430 ± 0.033	0.432 ± 0.032 ($\gamma = 0.02$)	0.425 ± 0.033	0.432 ± 0.033
Body height	0.388 ± 0.037	0.388 ± 0.037	0.389 ± 0.038 ($\gamma = 0.02$)	0.387 ± 0.037	0.387 ± 0.037
Length/height	0.274 ± 0.038	0.273 ± 0.038	0.278 ± 0.036 ($\gamma = 0.05$)	0.275 ± 0.037	0.277 ± 0.038

Table 2. Predictive abilities of various methods for four traits in large yellow croaker. [a]The optimized result estimated by FMixP when γ equals the value in the parentheses.

Computation time (minute)	BayesA	BayesCπ	MMixP	FMixP ($\gamma = 0.05$)	FMixP ($\gamma = 0.5$)
Simulated data	309.6	268.1	428.7	0.48	1.8
length/height	210.5	229.2	263.1	0.02	0.02

Table 3. Computation time of genomic prediction using various Bayesian methods for trait length/height.

as γ is larger than 0.05 or even 0.02. The values of γ estimated by MMixP are 0.28, 0.32, 0.27 and 0.31 for the traits body weight, body length, body height and length/height, respectively.

Computation time. Table 3 shows the computation time of each method for the simulated data and the trait length/height in large yellow croaker. The Fortran90 codes were run in a computer with an Intel Xeon CPU E7-4820. The computation time of MMixP is the longest in all statistical methods. Compared with the BayesCπ, the computational speed of BayesA is slightly slower in the simulated data but slightly faster in the real data. However, all MCMC-based Bayesian methods show a much slower computational speed than FMixP. The computation time for FMixP with $\gamma = 0.5$ is longer than that for FMixP with $\gamma = 0.05$ in the simulated data, but this difference is not obvious in the real data. We also compared the computation time between MixP introduced by

Yu and Meuwissen and our FMixP, and the results showed that the time of their MixP was approximately 20~25% longer than that of our FMixP.

Discussion

In this study, we compared the predictive abilities among BayesA, BayesCπ, MMixP and FMixP. When $\gamma = 0.5$, the results of FMixP are equivalent to those of GBLUP or RRBLUP, an observation which was also mentioned by Yu and Meuwissen[9]. Hence, the predictive result of FMixP when $\gamma = 0.5$ is the same as that of GBLUP in Shepherd *et al.*[8], in which the same simulated data was used. Therefore, we actually compared the results of five methods (i.e., BayesA, BayesCπ, MMixP, FMixP and GBLUP) in this study. The results show that the ranking of the predictive results among the different methods is not consistent between the simulated and real data. In the simulated data, the ranking according to predictive accuracy is: BayesCπ ≈ MMixP ≈ FMixP ($\gamma = 0.07$) > BayesA > GBLUP. However, all of the methods yield almost the same result within a given trait in real data from large yellow croaker. A reasonable explanation may be that there is a small number of QTLs in the simulated data but many more QTLs in the real data. There are two reasons that support this speculation: (i) The simulated results of Yu and Meuwissen showed that accuracy was not sensitive to γ when the number of QTL loci was large, but FMixP with $\gamma < 0.5$ performed better than GBLUP if there was a small number of QTLs[9]. The results shown in Figs 2 and 4 are consistent with the above two cases. (ii) The values of γ estimated by MMixP in simulated data are much lower than that estimated in the real data, indicating there may be many QTL loci affecting the phenotypes in large yellow croaker. Another explanation is that when the LD between markers is not strong, the accuracy may be due to the relationships captured by markers[12,13]. In this case, the GBLUP and various Bayesian methods may yield similar predictive results.

In addition to the predictive accuracy, computational speed is another important aspect in genomic prediction. This study shows that FMixP is significantly faster than the MCMC-based Bayesian methods. The main reason for this difference is that FMixP is not based on MCMC algorithms which are sampling processes and require many cycles to obtain a precise solution. It shows that the computation time for BayesCπ is slightly longer than BayesA in the real data, but slightly shorter in the simulated data. This is because the computational speed of BayesCπ is based on the value of π. A smaller π means more SNPs have zero effects and thus do not need to be sampled from the posterior normal distribution. MMixP needs more computation time than BayesA and BayesCπ, because there are more variables that need to be sampled in MMixP. For example, the SNP effect with variance equalling zero is not sampled in BayesCπ. However, all SNP effects need to be sampled in each Gibbs sampling cycle in MMixP, because each SNP may have a large or small variance. The computational speed for FMixP with $\gamma = 0.05$ is faster than for FMixP with $\gamma = 0.5$ in the simulated data. The possible reason for this is that the number of QTLs is very small in the simulated data. FMixP with $\gamma = 0.05$ is closer to the real QTL distribution, so that FMixP with $\gamma = 0.05$ has a faster convergence speed. This also suggests that there may be more QTL loci in the real data, because there is no obvious difference in computation time for FMixP with $\gamma = 0.05$ or 0.5.

This study proposed two new computing strategies: one strategy for FMixP and the other one for MMixP. Compared with the derivation of Yu and Meuwissen[9], we used a simpler derivation to obtain the solutions in FMixP. The advantage of FMixP is the extremely fast computational speed. However, the probability of a SNP having a large variance (represented as γ) and variances of SNP effects cannot be estimated by this implementation. Instead, using the MCMC algorithm can estimate the γ and various variances, but the computational speed is significantly slower than FMixP. The two strategies may provide some references to others who want to perform genomic prediction in the future.

Material and Methods

Ethics approval. This study and all experimental protocols were approved by the Animal Care and Use Committee of the Fisheries College of Jimei University (Animal Ethics no. 1067). All methods were performed in accordance with approved guidelines.

Analytical derivation for FMixP. The linear model for genomic prediction was as follows:

$$\mathbf{y} = \mathbf{Xu} + \mathbf{Bg} + \mathbf{e}, \tag{2}$$

where $\mathbf{y}$ is a vector of phenotypic records, $\mathbf{X}$ is the design matrix for fixed effects, and $\mathbf{u}$ is a vector of fixed effects. In the simulated data, $\mathbf{X} = (1\ 1\ \ldots\ 1)'$ and $\mathbf{u}$ is overall mean, whereas in the real data, the fixed effects were the sexual effects, $\mathbf{X}_i = (1\ 0)$ for male and $(0\ 1)$ for female. $\mathbf{B}$ is the matrix of SNP genotypes (coded as 0 for genotype '*A*_*A*', 1 for '*A*_*a*' and 2 for '*a*_*a*'), $\mathbf{g}$ is a vector of SNP effects, and $\mathbf{e}$ is a vector of residual effects, where $\mathbf{e} \sim N(0, \mathbf{I}\sigma_e^2)$. Genotypic codes were standardised using the formula: $B'_{ij} = \left(B_{ij} - 2p_j\right)/\sqrt{2p_j(1-p_j)}$, where p_j is the frequency of allele '*a*' at locus *j*.

In this study, the prior distribution was the same as that described by Yu and Meuwissen[9]. According to the prior distribution for SNP variance, the prior for SNP effect g_j can be written as a mixture of normal distributions:

$$\pi\left(g_j\right) = \gamma\phi\left(g_j|0,\ \sigma_1^2\right) + (1-\gamma)\phi\left(g_j|0,\ \sigma_2^2\right), \tag{3}$$

where g_j is the effect of SNP *j*.

Here, we used an Iterative Conditional Expectation (ICE) algorithm[7] to estimate the SNP effects. This algorithm estimates $E(\mathbf{g}|\mathbf{y})$ for each SNP effect in turn, where the current effects of the other SNPs are assumed to be known values. For example, if $E(g_j|\mathbf{y}_{-j})$ is estimated, the current effects of all other SNPs are used to calculate the $\mathbf{y}_{-j}$, i.e.,

$$\mathbf{y}_{-j} = \mathbf{y} - \mathbf{Xu} - \sum_{k \neq j} \mathbf{B}_k g_k, \tag{4}$$

where $\mathbf{B}_k$ is a vector from the k^{th} column of $\mathbf{B}$. The expectation of SNP effect, $E(g_j|\mathbf{y}_{-j})$, is estimated by a Bayesian model[7,9]:

$$\begin{aligned} E\left(g_j \middle| \mathbf{y}_{-j}\right) &= \int_{-\infty}^{+\infty} g_j \, f\left(g_j \middle| \mathbf{y}_{-j}\right) dg_j \\ &= \frac{\int_{-\infty}^{+\infty} g_j \, f\left(\mathbf{y}_{-j} \middle| \mathbf{B}_j g_j, \mathbf{I}\sigma_e^2\right) \pi\left(g_j\right) dg_j}{\int_{-\infty}^{+\infty} f\left(\mathbf{y}_{-j} \middle| \mathbf{B}_j g_j, \mathbf{I}\sigma_e^2\right) \pi\left(g_j\right) dg_j} \end{aligned} \tag{5}$$

where the $f(\mathbf{y}_{-j}|\mathbf{B}_j g_j, \mathbf{I}\sigma_e{}^2)$ is a multivariate normal density. Evaluating this multivariate density will be computationally intense because it involves calculating the determinant and inverse of variance-covariance matrix for the data $\mathbf{y}_{-j}$. However, the $f(\mathbf{y}_{-j}|\mathbf{B}_j g_j, \mathbf{I}\sigma_e^2)$ is proportional to the product of univariate normal densities $f(Y|g_j, \sigma^2)$, where $Y = (\mathbf{B}_j'\mathbf{B}_j)^{-1}\mathbf{B}_j'\mathbf{y}_{-j}$ and $\sigma^2 = (\mathbf{B}_j'\mathbf{B}_j)^{-1}\sigma_e^2$ (See Appendix 2 of Meuwissen *et al.*[7]). Unlike the derivation of Yu and Meuwissen[9], we did not calculate the multivariate likelihood but simplified the derivation using $f(Y|g_j, \sigma^2)$ to replace $f(\mathbf{y}_{-j}|\mathbf{B}_j g_j, \mathbf{I}\sigma_e{}^2)$. Thus, the equation (5) can be rewritten as:

$$E\left(g_j \middle| \mathbf{y}_{-j}\right) = \frac{\int_{-\infty}^{+\infty} g_j \, f\left(Y \middle| g_j, \sigma^2\right) \pi\left(g_j\right) dg_j}{\int_{-\infty}^{+\infty} f\left(Y \middle| g_j, \sigma^2\right) \pi\left(g_j\right) dg_j}. \tag{6}$$

Combined with equation (3), the numerator of equation (6) can be split into two terms:

$$\gamma \int_{-\infty}^{+\infty} g_j \, f\left(Y \middle| g_j, \sigma^2\right) \phi\left(g_j \middle| 0, \sigma_1^2\right) dg_j + (1-\gamma) \int_{-\infty}^{+\infty} g_j f\left(Y \middle| g_j, \sigma^2\right) \phi\left(g_j \middle| 0, \sigma_2^2\right) dg_j. \tag{7}$$

The first term in formula (7) can be derived as follows:

$$\begin{aligned} &\gamma \int_{-\infty}^{+\infty} g_j f\left(Y \middle| g_j, \sigma^2\right) \phi\left(g_j \middle| 0, \sigma_1^2\right) dg_j \\ &= \frac{\gamma}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} \frac{g_j}{\sqrt{2\pi}\sigma\sigma_1} \exp\left[-\frac{\left(Y - g_j\right)^2}{2\sigma^2} - \frac{g_j^2}{2\sigma_1^2} \right] dg_j \\ &= \frac{\gamma}{\sqrt{2\pi}} \exp\left[-\frac{Y^2}{2(\sigma_1^2 + \sigma^2)} \right] \frac{1}{\sqrt{\sigma_1^2 + \sigma^2}} \int_{-\infty}^{+\infty} \frac{g_j}{\sqrt{2\pi}\frac{\sigma\sigma_1}{\sqrt{\sigma_1^2 + \sigma^2}}} \exp\left[-\frac{\left(g_j - \frac{Y\sigma_1^2}{\sigma_1^2 + \sigma^2}\right)^2}{2\left(\frac{\sigma\sigma_1}{\sqrt{\sigma_1^2 + \sigma^2}}\right)^2} \right] dg_j. \end{aligned} \tag{8}$$

The last term in formula (8) can be taken as calculating the expected value of g_j in the normal distribution with a mean $Y\sigma_1{}^2/(\sigma_1{}^2 + \sigma^2)$ and variance $\sigma^2\sigma_1{}^2/(\sigma_1{}^2 + \sigma^2)$, so this term equals $Y\sigma_1{}^2/(\sigma_1{}^2 + \sigma^2)$. Thus, the first term of formula (7) becomes:

$$\frac{\gamma}{\sqrt{2\pi}} \exp\left[-\frac{Y^2}{2(\sigma_1^2 + \sigma^2)} \right] \frac{1}{\sqrt{\sigma_1^2 + \sigma^2}} \frac{Y\sigma_1^2}{\sigma_1^2 + \sigma^2}. \tag{9}$$

Similarly, the second term becomes:

$$\frac{1-\gamma}{\sqrt{2\pi}} \exp\left[-\frac{Y^2}{2(\sigma_2^2 + \sigma^2)} \right] \frac{1}{\sqrt{\sigma_2^2 + \sigma^2}} \frac{Y\sigma_2^2}{\sigma_2^2 + \sigma^2}. \tag{10}$$

Thus, the numerator of equation (6) equals:

$$\frac{\gamma}{\sqrt{2\pi}} \exp\left[-\frac{Y^2}{2(\sigma_1^2 + \sigma^2)} \right] \frac{1}{\sqrt{\sigma_1^2 + \sigma^2}} \frac{Y\sigma_1^2}{\sigma_1^2 + \sigma^2} + \frac{1-\gamma}{\sqrt{2\pi}} \exp\left[-\frac{Y^2}{2(\sigma_2^2 + \sigma^2)} \right] \frac{1}{\sqrt{\sigma_2^2 + \sigma^2}} \frac{Y\sigma_2^2}{\sigma_2^2 + \sigma^2}. \tag{11}$$

The derivation of the denominator in equation (6) is very similar to that of the numerator, but there is no g_j in the integrand. Therefore, the integral is not to calculate the expected value, but rather to calculate the cumulative probability from $-\infty$ to ∞, so this value is 1. Thus, the denominator in equation (6) can be written as:

$$\frac{\gamma}{\sqrt{2\pi}}\exp\left[-\frac{Y^2}{2(\sigma_1^2+\sigma^2)}\right]\frac{1}{\sqrt{\sigma_1^2+\sigma^2}}+\frac{1-\gamma}{\sqrt{2\pi}}\exp\left[-\frac{Y^2}{2(\sigma_2^2+\sigma^2)}\right]\frac{1}{\sqrt{\sigma_2^2+\sigma^2}}. \quad (12)$$

Thus, we derive the final form for equation (6),

$$E\left(g_j\middle|\mathbf{y}_{-j}\right)=\frac{\gamma\frac{Y\sigma_1^2}{\sigma_1^2+\sigma^2}+(1-\gamma)\exp\left[\frac{1}{2}\left(\frac{Y^2}{\sigma_1^2+\sigma^2}-\frac{Y^2}{\sigma_2^2+\sigma^2}\right)\right]\frac{\sqrt{\sigma_1^2+\sigma^2}}{\sqrt{\sigma_2^2+\sigma^2}}\frac{Y\sigma_2^2}{\sigma_2^2+\sigma^2}}{\gamma+(1-\gamma)\exp\left[\frac{1}{2}\left(\frac{Y^2}{\sigma_1^2+\sigma^2}-\frac{Y^2}{\sigma_2^2+\sigma^2}\right)\right]\frac{\sqrt{\sigma_1^2+\sigma^2}}{\sqrt{\sigma_2^2+\sigma^2}}}. \quad (13)$$

The fixed effects are estimated in each iteration by the formula: $\hat{\mathbf{u}}=(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'(\mathbf{y}-\mathbf{B}\hat{\mathbf{g}})$. We judged the convergence of solutions at the tth iteration according to the formula $(\mathbf{G}^t-\mathbf{G}^{t-1})'(\mathbf{G}^t-\mathbf{G}^{t-1})/(\mathbf{G}^{t\prime}\mathbf{G}^t)<10^{-8}$, where $\mathbf{G}=(\hat{\mathbf{u}}'\ \hat{\mathbf{g}}')'$.

Derivation for MMixP. FMixP does not estimate the parameter γ, such that a direct search should be used to obtain the optimal value of γ in genomic prediction. However, the value of γ can be estimated by the MCMC algorithm. With MMixP, the prior distributions of various variables, such as γ, $\mathbf{u}$, $\mathbf{g}$, σ_1^2, σ_2^2 and σ_e^2, are required. The priors for γ, $\mathbf{u}$ and σ_e^2 were assumed to follow uniform distributions. The prior for g_j depended on the γ and variances:

$$g_j|\gamma,\sigma^2\sim\begin{cases}N(0,\sigma_1^2) & \text{with probability } \gamma\\ N(0,\sigma_2^2) & \text{with probability } (1-\gamma)\end{cases}, \quad (14)$$

where γ is the probability that a SNP has a large variance, and σ_1^2 and σ_2^2 represent the large and small variance, respectively. The priors of σ_1^2 and σ_2^2 were assumed to follow the inverse-chi-squared distributions:

$$\begin{cases}\sigma_1^2\sim\chi^{-2}(\nu,s_1^2) & \text{where } s_1^2=\dfrac{(\nu-2)(1-\gamma)V_g}{\nu\gamma M}\\ \sigma_2^2\sim\chi^{-2}(\nu,s_2^2) & \text{where } s_2^2=\dfrac{(\nu-2)\gamma V_g}{\nu(1-\gamma)M}\end{cases}, \quad (15)$$

The scale parameter s_1^2 was set because $E(\sigma_1^2)=\frac{\nu s_1^2}{\nu-2}=\frac{(1-\gamma)V_g}{\gamma M}$ according to the properties of inverse-chi-squared distribution and Pareto principle. A similar method was used to set the parameter s_2^2. An indicator variable δ_j was used to indicate whether SNP j had a large or small variance. The prior for δ_j was $p(\delta_j|\gamma)=\gamma^{\delta_j}(1-\gamma)^{(1-\delta_j)}$, where $\delta_j=1$ and $\delta_j=0$ represent the $\sigma_j^2=\sigma_1^2$ with probability γ and $\sigma_j^2=\sigma_2^2$ with probability $(1-\gamma)$, respectively.

The δ_j and g_j are sampled from their joint conditional distribution, because the sampling strategy of g_j is dependent on the value of δ_j. The joint conditional distribution can be written as:

$$f\left(\delta_j,g_j\middle|\mathbf{y},\mathbf{u},\mathbf{g}_{-j},\boldsymbol{\delta}_{-j},\boldsymbol{\sigma}^2,\sigma_e^2,\gamma\right)=f\left(\delta_j\middle|\mathbf{y},\mathbf{u},\mathbf{g}_{-j},\boldsymbol{\delta}_{-j},\boldsymbol{\sigma}^2,\sigma_e^2,\gamma\right)f\left(g_j\middle|\delta_j,\mathbf{y},\mathbf{u},\mathbf{g}_{-j},\boldsymbol{\delta}_{-j},\boldsymbol{\sigma}^2,\sigma_e^2,\gamma\right), \quad (16)$$

where $\mathbf{g}_{-j}$ and $\boldsymbol{\delta}_{-j}$ represent the vectors of SNP effects and indicator variables except g_j and δ_j, respectively, and $\boldsymbol{\sigma}^2=(\sigma_1^2,\sigma_2^2)$. Then the conditional distribution for δ_j can be written as:

$$\begin{aligned}f\left(\delta_j\middle|\mathbf{y},\mathbf{u},\mathbf{g}_{-j},\boldsymbol{\delta}_{-j},\boldsymbol{\sigma}^2,\sigma_e^2,\gamma\right)&\propto f\left(\mathbf{y}\middle|\delta_j,\mathbf{u},\mathbf{g}_{-j},\boldsymbol{\delta}_{-j},\boldsymbol{\sigma}^2,\sigma_e^2,\gamma\right)p\left(\delta_j|\gamma\right)\\&\propto f\left(\mathbf{y}_{-j}\middle|\delta_j,\sigma_j^2,\sigma_e^2\right)p\left(\delta_j|\gamma\right)\end{aligned}, \quad (17)$$

where $\mathbf{y}_{-j}=\mathbf{y}-\mathbf{X}\mathbf{u}-\sum_{k\neq j}\mathbf{B}_k g_k=\mathbf{B}_j g_j+\mathbf{e}$, as in equation (4). Thus, $f\left(\mathbf{y}_{-j}\middle|\delta_j,\sigma_j^2,\sigma_e^2\right)p\left(\delta_j|\gamma\right)$ can be represented as:

$$f\left(\mathbf{y}_{-j}\middle|\delta_j,\sigma_j^2,\sigma_e^2\right)p\left(\delta_j|\gamma\right)=\begin{cases}f\left(\mathbf{y}_{-j}|\sigma_1^2,\sigma_e^2\right)\gamma & \text{when } \delta_j=1\\ f\left(\mathbf{y}_{-j}|\sigma_2^2,\sigma_e^2\right)(1-\gamma) & \text{when } \delta_j=0\end{cases}. \quad (18)$$

As $f(\delta_j=1|\mathbf{y},\mathbf{u},\mathbf{g}_{-j},\boldsymbol{\delta}_{-j},\boldsymbol{\sigma}^2,\sigma_e^2,\gamma)+f(\delta_j=0|\mathbf{y},\mathbf{u},\mathbf{g}_{-j},\boldsymbol{\delta}_{-j},\boldsymbol{\sigma}^2,\sigma_e^2,\gamma)=1$, $f(\delta_j=1|\mathbf{y},\mathbf{u},\mathbf{g}_{-j},\boldsymbol{\delta}_{-j},\sigma^2,\sigma_e^2,\gamma)$ can be sampled from:

$$
\begin{aligned}
&f\left(\delta_j = 1 \middle| \mathbf{y}, \mathbf{u}, \mathbf{g}_{-j}, \boldsymbol{\delta}_{-j}, \boldsymbol{\sigma}^2, \sigma_e^2, \gamma\right) \\
&= \frac{f\left(\delta_j = 1 \middle| \mathbf{y}, \mathbf{u}, \mathbf{g}_{-j}, \boldsymbol{\delta}_{-j}, \boldsymbol{\sigma}^2, \sigma_e^2, \gamma\right)}{f\left(\delta_j = 1 \middle| \mathbf{y}, \mathbf{u}, \mathbf{g}_{-j}, \boldsymbol{\delta}_{-j}, \boldsymbol{\sigma}^2, \sigma_e^2, \gamma\right) + f\left(\delta_j = 0 \middle| \mathbf{y}, \mathbf{u}, \mathbf{g}_{-j}, \boldsymbol{\delta}_{-j}, \boldsymbol{\sigma}^2, \sigma_e^2, \gamma\right)} \\
&= \frac{f\left(\mathbf{y}_{-j} | \sigma_1^2, \sigma_e^2\right)\gamma}{f\left(\mathbf{y}_{-j} | \sigma_1^2, \sigma_e^2\right)\gamma + f\left(\mathbf{y}_{-j} | \sigma_2^2, \sigma_e^2\right)(1-\gamma)} \\
&= \frac{1}{1 + \frac{f\left(\mathbf{y}_{-j} | \sigma_2^2, \sigma_e^2\right)(1-\gamma)}{f\left(\mathbf{y}_{-j} | \sigma_1^2, \sigma_e^2\right)\gamma}}.
\end{aligned} \tag{19}
$$

Note that $f(\mathbf{y}_{-j}|\sigma_j^2, \sigma_e^2)$ is a multivariate density, the case of which is similar to that in FMixP. An efficient way is to use the product of univariate distributions of $\mathbf{B}_j'\mathbf{y}_{-j}$ instead of the distribution of $\mathbf{y}_{-j}$[13,14]. The $f(\mathbf{B}_j'\mathbf{y}_{-j}|\sigma_j^2, \sigma_e^2)$ has zero mean and variance $(\mathbf{B}_j'\mathbf{B}_j)^2\sigma_j^2 + \mathbf{B}_j'\mathbf{B}_j\sigma_e^2$. Thus, the equation (19) can be written as:

$$
\begin{aligned}
&f\left(\delta_j = 1 \middle| \mathbf{y}, \mathbf{u}, \mathbf{g}_{-j}, \boldsymbol{\delta}_{-j}, \boldsymbol{\sigma}^2, \sigma_e^2, \gamma\right) \\
&= \frac{1}{1 + \frac{f\left(\mathbf{B}_j'\mathbf{y}_{-j} | \sigma_2^2, \sigma_e^2\right)(1-\gamma)}{f\left(\mathbf{B}_j'\mathbf{y}_{-j} | \sigma_1^2, \sigma_e^2\right)\gamma}} \\
&= \frac{1}{1 + \exp\left[0.5\log(V_1) - 0.5\log(V_2) + \frac{0.5\left(\mathbf{B}_j'\mathbf{y}_{-j}\right)^2}{V_1} - \frac{0.5\left(\mathbf{B}_j'\mathbf{y}_{-j}\right)^2}{V_2} + \log(1-\gamma) - \log(\gamma)\right]},
\end{aligned} \tag{20}
$$

where $V_1 = (\mathbf{B}_j'\mathbf{B}_j)^2\sigma_1^2 + \mathbf{B}_j'\mathbf{B}_j\sigma_e^2$ and $V_2 = (\mathbf{B}_j'\mathbf{B}_j)^2\sigma_2^2 + \mathbf{B}_j'\mathbf{B}_j\sigma_e^2$. After the δ_j has been updated, g_j is sampled as:

$$
f\left(g_j \middle| \delta_j, \mathbf{y}, \mathbf{u}, \mathbf{g}_{-j}, \boldsymbol{\delta}_{-j}, \boldsymbol{\sigma}^2, \sigma_e^2, \gamma\right) \sim \begin{cases} N\left(\frac{\mathbf{B}'_j\mathbf{y}_{-j}}{\mathbf{B}'_j\mathbf{B}_j + \sigma_e^2/\sigma_1^2}, \frac{\sigma_e^2}{\mathbf{B}'_j\mathbf{B}_j + \sigma_e^2/\sigma_1^2}\right) & \text{if } \delta_j = 1 \\ N\left(\frac{\mathbf{B}_j'\mathbf{y}_{-j}}{\mathbf{B}'_j\mathbf{B}_j + \sigma_e^2/\sigma_2^2}, \frac{\sigma_e^2}{\mathbf{B}'_j\mathbf{B}_j + \sigma_e^2/\sigma_2^2}\right) & \text{if } \delta_j = 0 \end{cases}. \tag{21}
$$

As the σ_1^2 appears only in its own prior and the normal distribution of g_j with $\delta_j = 1$, the posterior distribution of σ_1^2 can be derived as:

$$
\begin{aligned}
f(\sigma_1^2|\mathbf{y}, \mathbf{u}, \mathbf{g}, \boldsymbol{\delta}, \sigma_2^2, \sigma_e^2, \gamma) &\propto f(\sigma_1^2)\prod_{\delta_{j=1}} f(g_j|\sigma_1^2) \\
&\propto (\sigma_1^2)^{-\frac{k+\nu+2}{2}}\exp\left(-\frac{\nu s_1^2 + \sum_{\delta_{j=1}} g_j^2}{2\sigma_1^2}\right), \\
&\sim \chi^{-2}\left(k+\nu, \frac{\nu s_1^2 + \sum_{\delta_{j=1}} g_j^2}{k+\nu}\right)
\end{aligned} \tag{22}
$$

where k is the number of SNP loci with $\delta_j = 1$. Similarly, the posterior distribution of σ_2^2 follows the inverse-chi-squared distribution $\chi^{-2}\left(m+\nu, \frac{\nu s_2^2 + \sum_{\delta_{j=0}} g_j^2}{m+\nu}\right)$, where m is the number of SNP loci with $\delta_j = 0$.

The starting value of γ was set to 0.5, and the posterior probability is drawn from the Beta($k+1, m+1$):

$$
\begin{aligned}
f(\gamma|\mathbf{y}, \mathbf{u}, \mathbf{g}, \boldsymbol{\delta}, \boldsymbol{\sigma}^2, \sigma_e^2) &\propto f(\gamma)f(\boldsymbol{\delta}|\gamma) \\
&\propto \gamma^k(1-\gamma)^m.
\end{aligned} \tag{23}
$$

Note that if the sampling value of γ is larger than 0.5, we can switch the labels of the variance σ_1^2 and σ_2^2, and set value of γ to $1-\gamma$. The posterior distributions of fixed effect $\mathbf{u}$ and residual variance σ_e^2 are the same as BayesA, which has been described in many studies[1,13,15].

Genomic prediction by other approaches. Two other Bayesian methods, BayesA[1] and BayesCπ[3], were used for comparison with MixP. The prior distribution of variances of SNP effects in BayesA follows an inverse-chi-squared distribution, i.e., $\sigma_j^2 \sim \chi^{-2}(\nu, s^2)$[1,11]. In BayesCπ, SNPs with non-zero effects have a common

Trait	Male		Female		Heritability[b]
	Number	Mean[a] ± SE	Number	Mean[a] ± SE	(Mean ± SE)
Body weight	237	202.22 ± 5.01	263	247.41 ± 6.16	0.61 ± 0.11
Body length	237	227.19 ± 1.64	263	234.85 ± 1.79	0.59 ± 0.10
Body height	237	62.03 ± 0.53	263	66.61 ± 0.59	0.52 ± 0.11
Length/height	237	3.68 ± 0.01	263	3.54 ± 0.01	0.32 ± 0.10

Table 4. Statistical results of the phenotypic data for four quantitative traits in large yellow croaker. [a]The units are gram (g) for body weight, and millimetre (mm) for body length and body height.

variance that also follows an inverse-chi-squared distribution[3]. The degree of freedom (v) of the inverse-chi-squared distribution was set to 5.0. As the SNP genotypes had been standardised, parameter s^2 was set without $\sum 2p_j(1-p_j)$ in the denominator, which was different from the formula derived by Habier *et al.*[3] and Gianola *et al.*[16]. In this study, $s^2 = [(v-2)V_g]/(vM)$ in BayesA and $s^2 = [(v-2)V_g]/(\pi vM)$ in BayesCπ, where π represents the probability of a SNP with a non-zero effect and is estimated by the MCMC algorithm. V_g is total additive genetic variance which is estimated using the R-package "EMMREML" (Version 3.1) that is one of packages[17–22] used to estimate genetic parameters. Before V_g estimation, a genomic relationship matrix (**G** matrix) was calculated using the formula[2]: $\mathbf{G} = \frac{(\mathbf{B}-\mathbf{P})(\mathbf{B}-\mathbf{P})'}{2\sum p_j(1-p_j)}$, where the jth column of P is a vector of the frequency of allele 'a' at the jth locus, i.e., $\mathbf{P}_j = (p_j, p_j, \ldots, p_j)'$. Gibbs sampling was run for 20000 cycles, and the first 10000 cycles were discarded as burn in.

Simulated data. Both the simulated and real data were used to compare the predictive results of various statistical methods. The simulated data had been distributed to the participants of the QTLMAS XII workshop. The data was described in detail by Lund *et al.*[23] and a summary is given as follows. Through a simulation of a historic population of 50 generations, 4665 and 1200 individuals were simulated in the training and testing data sets, respectively. Six-thousand biallelic SNP loci were evenly spaced on 6 Morgan chromosomes, and 5,726 SNPs with minor allele frequencies (MAF) ≥0.05 were used for research. Forty-eight QTL loci were simulated, and the effects were sampled from a gamma distribution with a scale parameter 5.4 and a shape parameter of 0.42. The residual values were sampled to obtain a heritability value of 0.3 for the trait.

Real data on large yellow croaker. The experimental materials were large yellow croaker (*Larimichthys crocea*), which is one of the most commercially important marine fish species in southeast China and Eastern Asia[24]. All fish were reared in a breeding nucleus farm named 'Jinling Aquaculture Science and Technology Co. Ltd.' in Ningde City, Fujian Province, P.R. China. In total, 30 males and 30 females were mated randomly in a pool, and a total of 500 progenies (237 males and 263 females) were randomly selected and measured in the experiment. The trial was carried out in the Key Laboratory of Healthy Mariculture for the East China Sea when the fish were two years old. Four quantitative traits, body weight, body length, body height and the length/height ratio, were selected to perform genomic prediction. Growth rate and body shape (customers prefer purchasing fish with slender bodies) are the important traits for large yellow croaker, so these four traits were selected for research. The parameters of the four traits are shown in Table 4.

Next generation sequencing and genotyping. Fin samples from 500 individuals were collected for genotyping. The Genotyping-By-Sequencing (GBS) method was used to construct the libraries for next generation sequencing (NGS). Genomic DNA was incubated at 37 °C with *Eco*RI and *Nla*III, CutSmart™ buffer and MilliQ water. Digestion reactions were heat-inactivated at 65 °C for 20 minutes and the reaction system was held at 8 °C. The digested DNA was ligated to adapter sequences with CutSmart™ buffer, ATP, T4 DNA ligase, adapter mix and MilliQ water at 16 °C. The restriction-ligation reaction was also heat-inactivated at 65 °C for 20 minutes and the reaction system was held at 8 °C afterward. The PCR reaction was performed using diluted restriction-ligation samples, dNTP, Taq DNA polymerase (NEB) and IlluminaF and indexing primers. Fragments that were 200~300 bp in size were isolated using a Gel Extraction Kit (Qiagen). Then, pair-end sequencing was performed using an Illumina high-throughput sequencing platform. The raw sequencing reads were quality checked by FastQC[25]. The high-quality, filtered reads were mapped to the large yellow croaker reference genome sequence by BWA version 0.7.10[26]. The alignment files were then sorted and the duplicates marked by Picard (http://picard.sourceforge.net). Then, the GATK package[27] was applied for SNP calling. As a result, 29,748 SNPs with a missing rate ≤20%, a MAF (minor allele frequency) ≥0.05 and genotypes in Hardy-Weinberg equilibrium were selected for further analysis. Beagle Version 3.3.2 software was used to impute the missing SNPs[28].

Cross-validation. Genomic prediction by a replicated training-testing method was used to evaluate the predictive results of the real data. Cross-validation of 10 replicates was performed. All 500 individuals were randomly and evenly divided into 10 groups of 50 individuals each. In each replicate, one of the groups was selected as the testing data set while the remaining nine groups were used as the training data set. To observe the relationship between the predictive results of FMixP with γ, we varied the value of γ from 0.01 to 0.5 (50 levels were used with 0.01 as a step size).

Predictive accuracy and predictive ability. In the simulation, the correlation coefficient between true genetic values and predicted genetic values, $r_{(TBV,GEBV)}$ was used to measure the predictive accuracy[1], where

GEBV = $\mathbf{B}\hat{\mathbf{g}}$, $\hat{\mathbf{g}}$ is the vector of estimated SNP effects and **B** is the SNP genotypes; and an individual true breeding value (TBV) can be obtained by summing up all simulated QTL effects. We give a brief explanation below. If an individual GEBV is close to its TBV, the predictive accuracy is high. But if one aims to assess the predictive accuracies of a set of GEBV, one can use $r_{(\mathrm{TBV},\mathrm{GEBV})}$, and higher $r_{(\mathrm{TBV},\mathrm{GEBV})}$ suggests higher predictive accuracy.

In the real data analysis, because the true breeding values are unknown, we used the predictive ability to measure the predictive accuracy, which is described as the correlation coefficient between **GEBV**, and the phenotypes adjusted for the covariates ($\mathbf{y} - \mathbf{X}\hat{\mathbf{u}}$, where only genetic and residual effects are left), $r_{(\mathbf{y}-\mathbf{X}\hat{\mathbf{u}},\ \mathbf{GEBV})}$[29]. The higher correlation between them is, the higher genetic variance captured by the genetic SNPs is, leading to higher predictive ability.

All 500 individuals were added to the prediction model to estimate the computation time for various Bayesian methods. All of the calculation processes (except the REML process) were implemented in Fortran90 codes and run on the computer server of Jimei University.

Availability of data. Raw DNA sequencing reads were deposited in NCBI with the project accession PRJNA309464 and SRA accession SRR3114179.

References

1. Meuwissen, T. H., Hayes, B. J. & Goddard, M. E. Prediction of total genetic value using genome-wide dense marker maps. *Genetics.* **157**, 1819–1829 (2001).
2. Vanraden, P. M. Efficient methods to compute genomic predictions. *J Dairy SCI.* **91**, 4414–4423 (2008).
3. Habier, D., Fernando, R. L., Kizilkaya, K. & Garrick, D. J. Extension of the bayesian alphabet for genomic selection. *Bmc Bioinformatics.* **12**, 1–12 (2011).
4. Campos, G. D. L. *et al*. Predicting quantitative traits with regression models for dense molecular markers and pedigree. *Genetics.* **182**, 375–385 (2009).
5. Mutshinda, C. M. & Sillanpaa, M. J. Extended Bayesian LASSO for multiple quantitative trait loci mapping and unobserved phenotype prediction. *Genetics.* **186**, 1067–1075 (2010).
6. Yi, N., George, V. & Allison, D. B. Stochastic search variable selection for identifying multiple quantitative trait loci. *Genetics.* **164**, 1129–1138 (2003).
7. Meuwissen, T. H., Solberg, T. R., Shepherd, R. & Woolliams, J. A. A fast algorithm for BayesB type of prediction of genome-wide estimates of genetic value. *Genet Sel Evol.* **41**, 1–10 (2009).
8. Shepherd, R. K., Meuwissen, T. H. & Woolliams, J. A. Genomic selection and complex trait prediction using a fast EM algorithm applied to genome-wide markers. *Bmc Bioinformatics.* **11**, 2568 (2010).
9. Yu, X. & Meuwissen, T. H. Using the Pareto principle in genome-wide breeding value estimation. *Genet Sel Evol.* **43**, 1–7 (2011).
10. Ronen, B. The Pareto managerial principle: When does it apply? *Int J Prod Res.* **45**, 2317–2325 (2007).
11. Wang, C. S., Rutledge, J. J. & Gianola, D. Marginal inferences about variance components in a mixed linear model using Gibbs sampling. *Genet Sel Evol.* **25**, 41–62 (1993).
12. Habier, D., Fernando, R. L. & Dekkers, J. C. The impact of genetic relationship information on genome-assisted breeding values. *Genetics.* **177**, 2389–2397 (2007).
13. Fernando, R. L. & Garrick, D. Bayesian methods applied to GWAS. *Methods in Molecular Biology.* **1019**, 237–274 (2013).
14. Cheng, H., Long, Q., Garrick, D. J. & Fernando, R. L. A fast and efficient Gibbs sampler for BayesB in whole-genome analyses. *Genet Sel Evol.* **47**, 1–7 (2015).
15. Campos, G. D. L., Hickey, J. M., Pong-Wong, R., Daetwyler, H. D. & Calus, M. P. L. Whole genome regression and prediction methods applied to plant and animal breeding. *Genetics.* **193**, 327–345 (2013).
16. Gianola, D., Campos, G. D. L., Hill, W. G., Manfredi, E. & Fernando, R. Additive genetic variability and the Bayesian alphabet. *Genetics.* **183**, 347–363 (2009).
17. Wang, C. *et al*. GVCBLUP: A computer package for genomic prediction and variance component estimation of additive and dominance effects. *Bmc Bioinformatics.* **15**, 1–9 (2014).
18. Kang, H. M. *et al*. Efficient control of population structure in model organism association mapping. *Genetics.* **178**, 1709–1723 (2008).
19. Lee, S. H. & van der Werf, J. H. MTG2: An efficient algorithm for multivariate linear mixed model analysis based on genomic information. *Bioinformatics.* **32**, 1420–1422 (2016).
20. Covarrubias-Pazaran, G. Genome-Assisted prediction of quantitative traits using the r package sommer. *Plos One.* **11**, e156744 (2016).
21. Yang, J., Lee, S. H., Goddard, M. E. & Visscher, P. M. GCTA: A tool for genome-wide complex trait analysis. *Am J Hum Genet.* **88**, 76–82 (2011).
22. Gilmour, A. R. *et al*. ASReml user guide release 1.0. *University of Hamburg Department for.* **104**, 20617–20637 (2009).
23. Lund, M. S., Sahana, G., Koning, D. J. D. & Su, G. Comparison of analyses of the QTLMAS XII common dataset. I: Genomic selection. *BMC Proceedings.* **3**(Suppl 1), S1 (2009).
24. Xiao, S. *et al*. Functional marker detection and analysis on a comprehensive transcriptome of large yellow croaker by next generation sequencing. *Plos One.* **10**, e124432 (2015).
25. Xi, Y. *et al*. HTQC: A fast quality control toolkit for Illumina sequencing data. *Bmc Bioinformatics.* **14**, 68–70 (2013).
26. Li, H. & Durbin, R. Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics.* **25**, 1754–1760 (2010).
27. Mckenna, A. *et al*. The Genome Analysis Toolkit: A MapReduce framework for analyzing next-generation DNA sequencing data. *Genome Res.* **20**, 1297–1303 (2010).
28. Browning, B. L. & Browning, S. R. A unified approach to genotype imputation and haplotype-phase inference for large data sets of trios and unrelated individuals. *Am J Hum Genet.* **84**, 210–223 (2009).
29. Legarra, A., Robert-Granié, C., Manfredi, E. & Elsen, J. M. Performance of genomic selection in mice. *Genetics.* **180**, 611–618 (2008).

Acknowledgements

This work was supported by the National Natural Science Foundation of China (U1205122), Key projects of the Xiamen Southern Ocean Research Centre (14GZY70NF34) and the Foundation for Innovation Research Team of Jimei University (2010A02). Shijun Xiao performed the SNP discovery. Kun Ye, Qingkai Chen, Junwei Chen, Yang Liu and other colleagues in the laboratory participated in fish sampling and traits measurement. We also thank the editors and reviewers for their many helpful suggestions for this article.

Author Contributions

Z.W. designed the experiments and revised the manuscript. L.D. performed the analyses and drafted the manuscript. M.F. revised the manuscript. All of the authors have read and approved the final manuscript.

Additional Information

Competing Interests: The authors declare that they have no competing interests.

Publisher's note: Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.