Supplementary Information for "Further Improvements to Linear Mixed Models for Genome-Wide Association Studies"

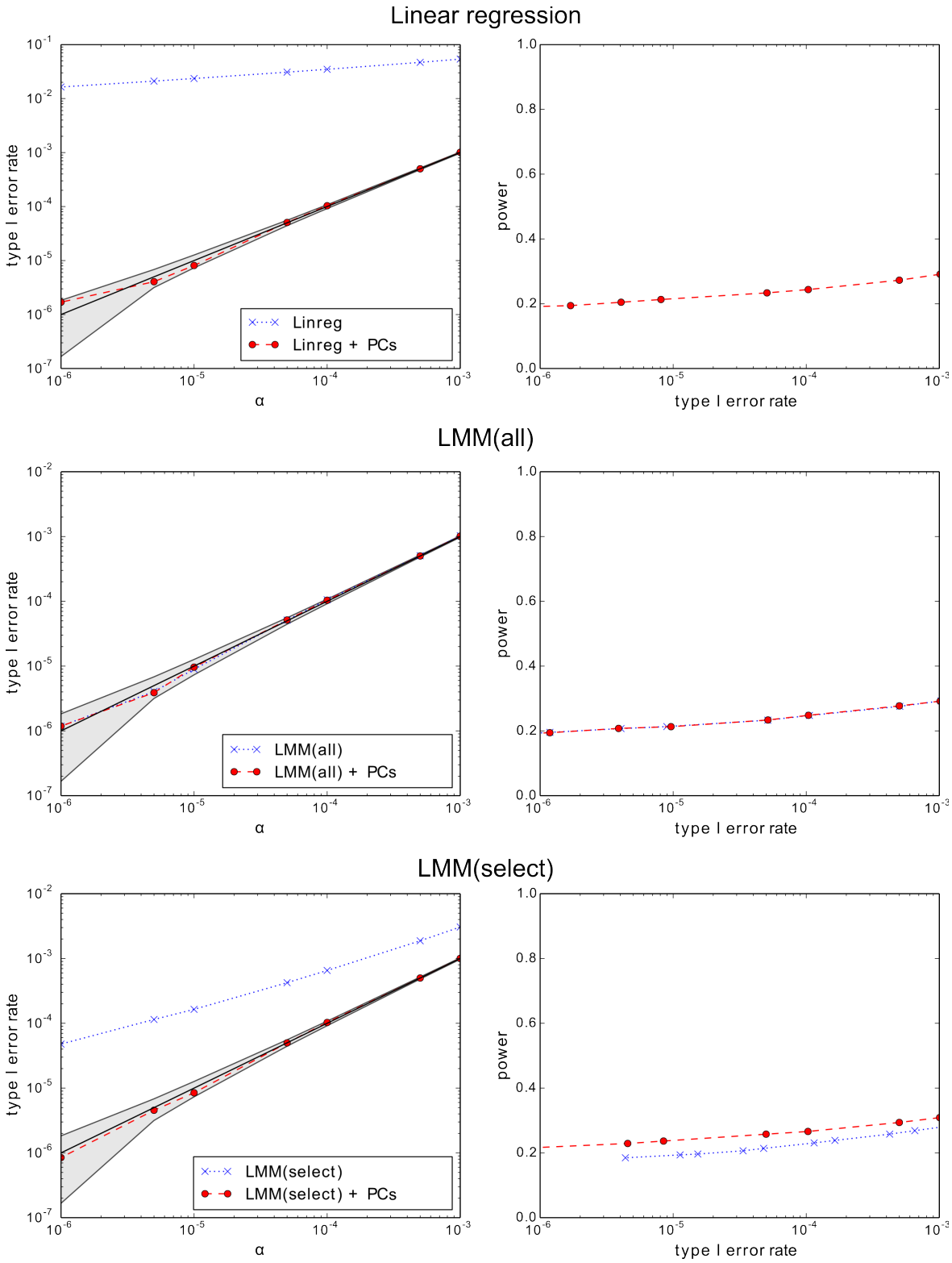
Christian Widmer[1*], Christoph Lippert[1*], Omer Weissbrod[2], Nicolo Fusi[1], Carl Kadie[3], Robert Davidson[3], Jennifer Listgarten[1], and David Heckerman[1*]

[1]eScience Group, Microsoft Research, 1100 Glendon Avenue, Suite PH1, Los Angeles, CA, 90024, United States

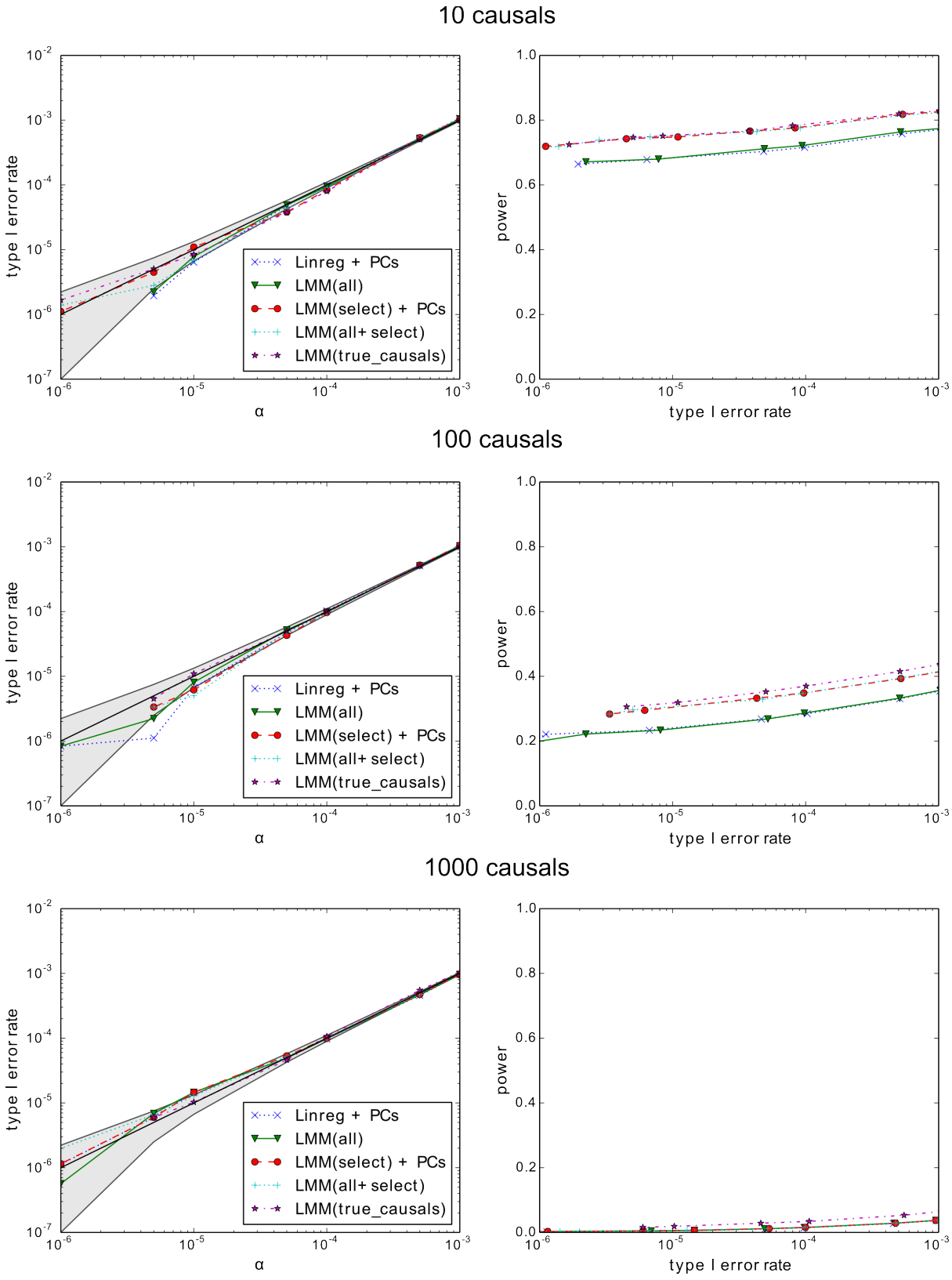
[2]Computer Science Department, Technion - Israel Institute of Technology, Haifa 32000, Israel

[3]eScience Group, Microsoft Research, One Microsoft Way, Redmond, WA, 98052, United States

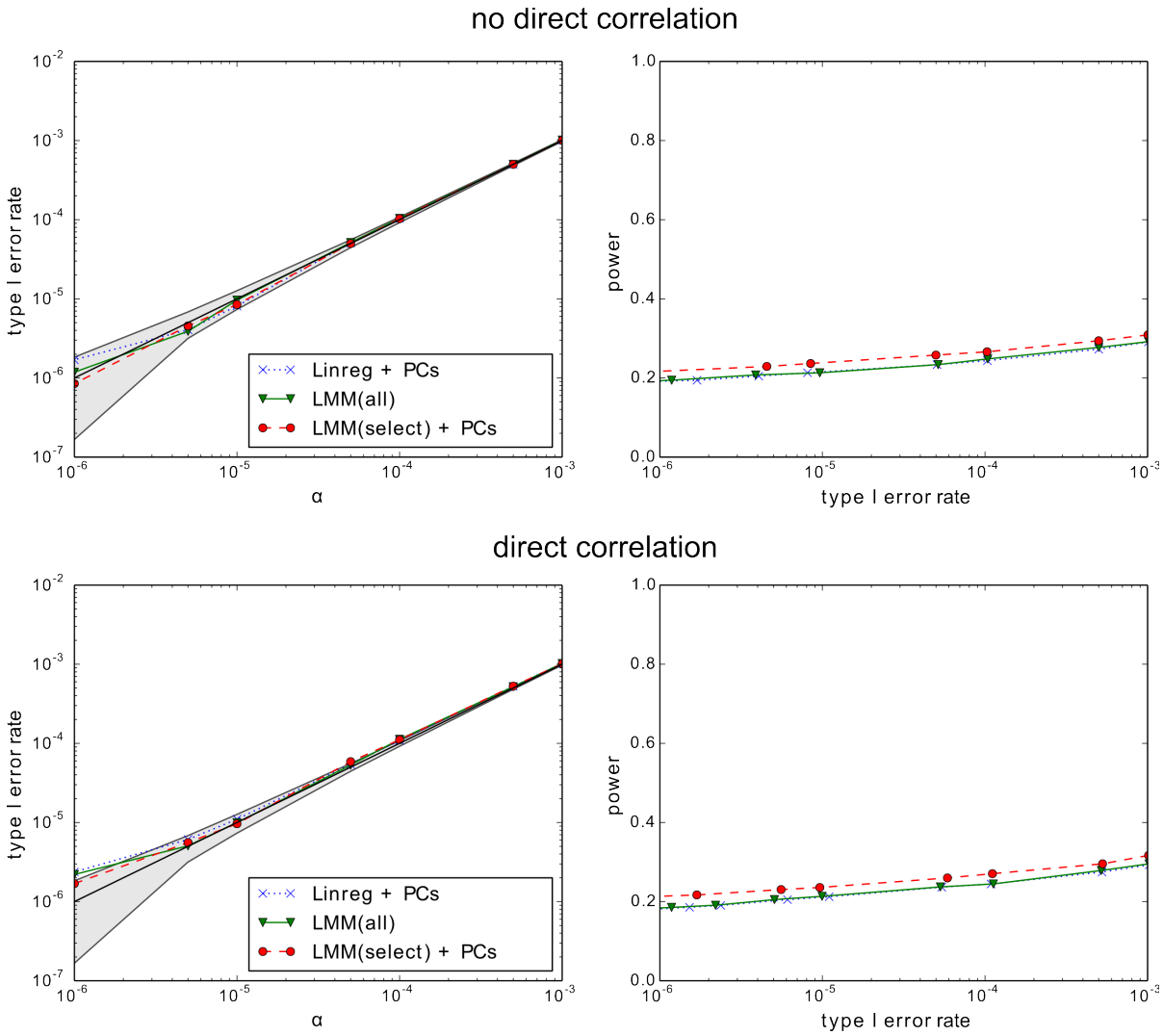
*These authors contributed equally. Please address correspondence to chwidmer@microsoft.com, lippert@microsoft.com, and heckerma@microsoft.com.



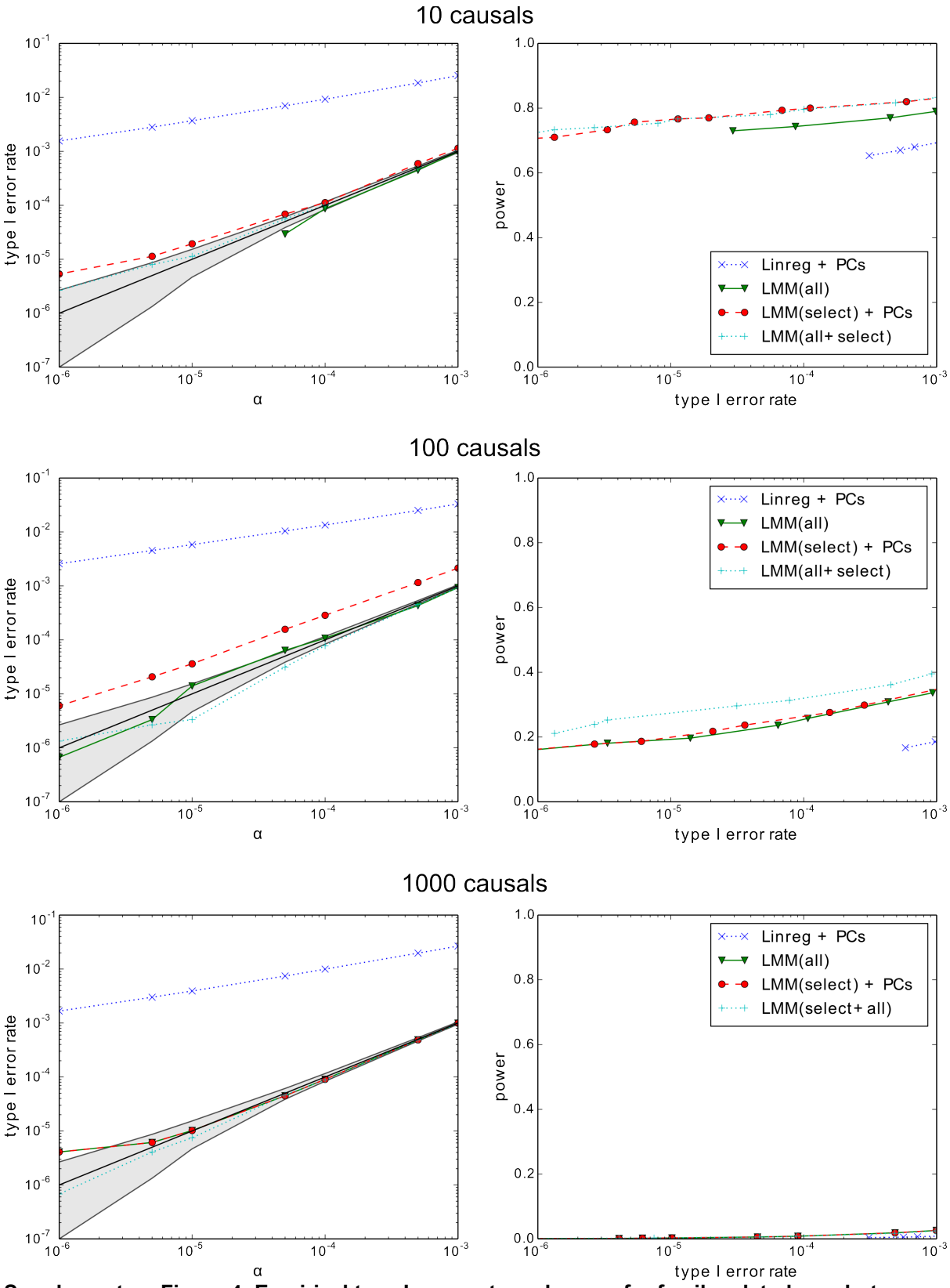
Supplementary Figure 1: Empirical type I error rate and power for population structure but no family relatedness with purely synthetic data. Each point represents the empirical type I error rate or power across 360 data sets with varying numbers of causal SNPs and with different degrees of signal (narrow-sense heritability) and population structure.



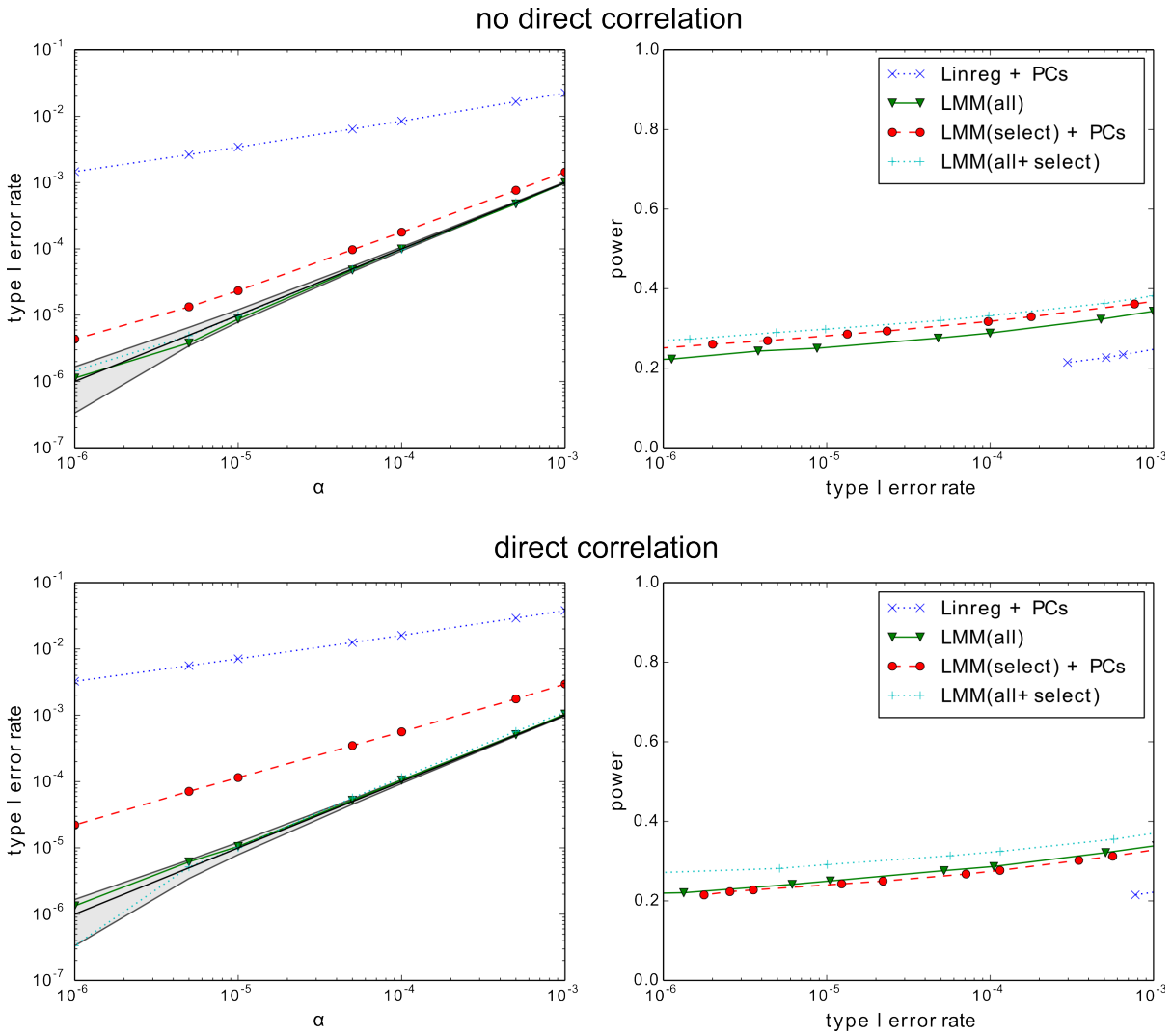
Supplementary Figure 2: Empirical type I error rate and power for population structure but no family relatedness with purely synthetic data. Each point represents the empirical type I error rate or power across 72 data sets with different degrees of signal (narrow-sense heritability) and population structure.



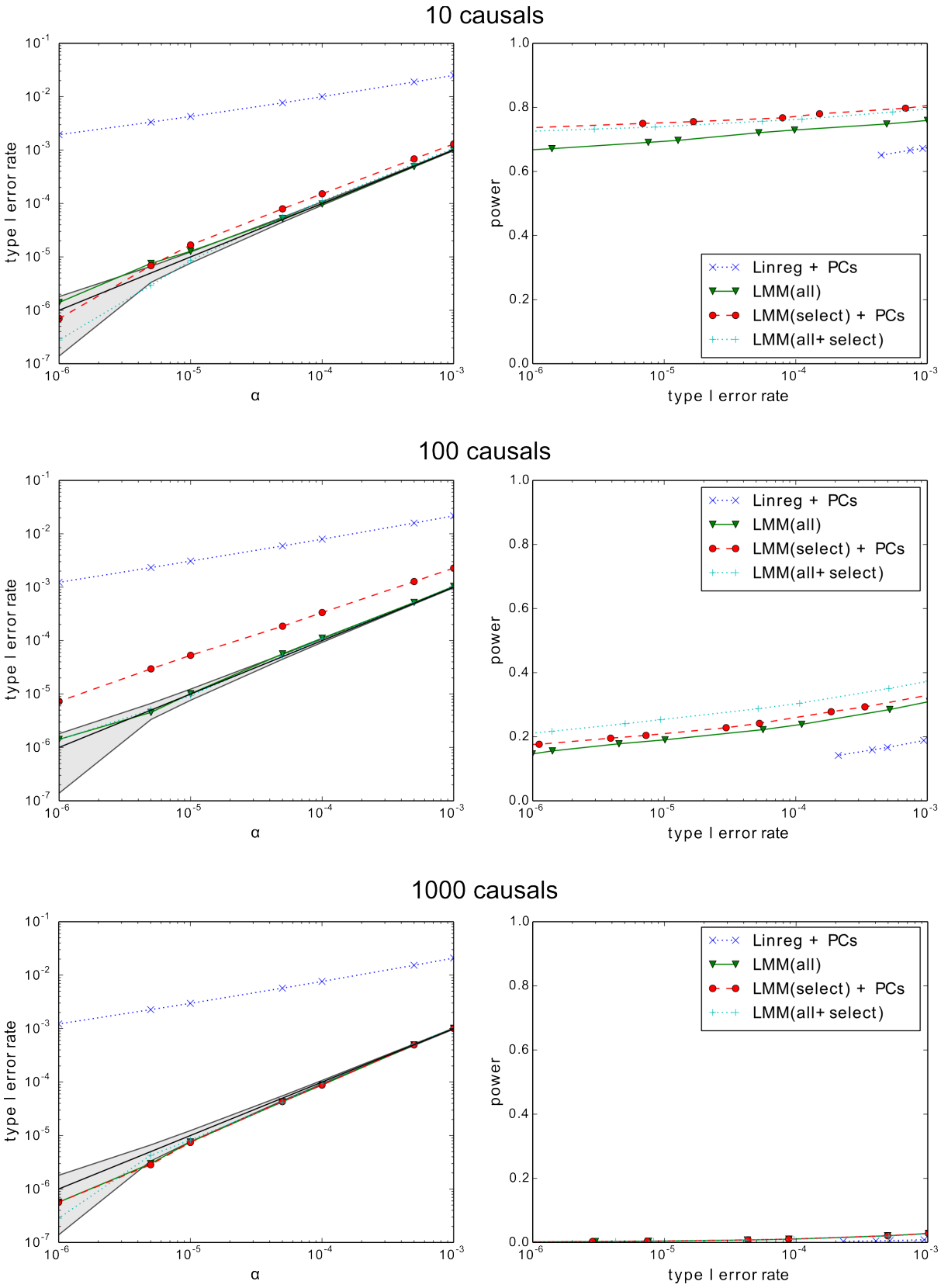
Supplementary Figure 3: Empirical type I error rate and power for population structure but no family relatedness, with and without a direct correlation between confounding structure and the phenotype. Each point represents the empirical type I error rate or power across 360 data sets with varying numbers of causal SNPs and with different numbers of causal SNPs and different degrees of signal (narrow-sense heritability) and population structure. The plots labeled “no direct correlation” and “direct correlation” correspond to the generating processes in **Figures 3a and 3b**, respectively.



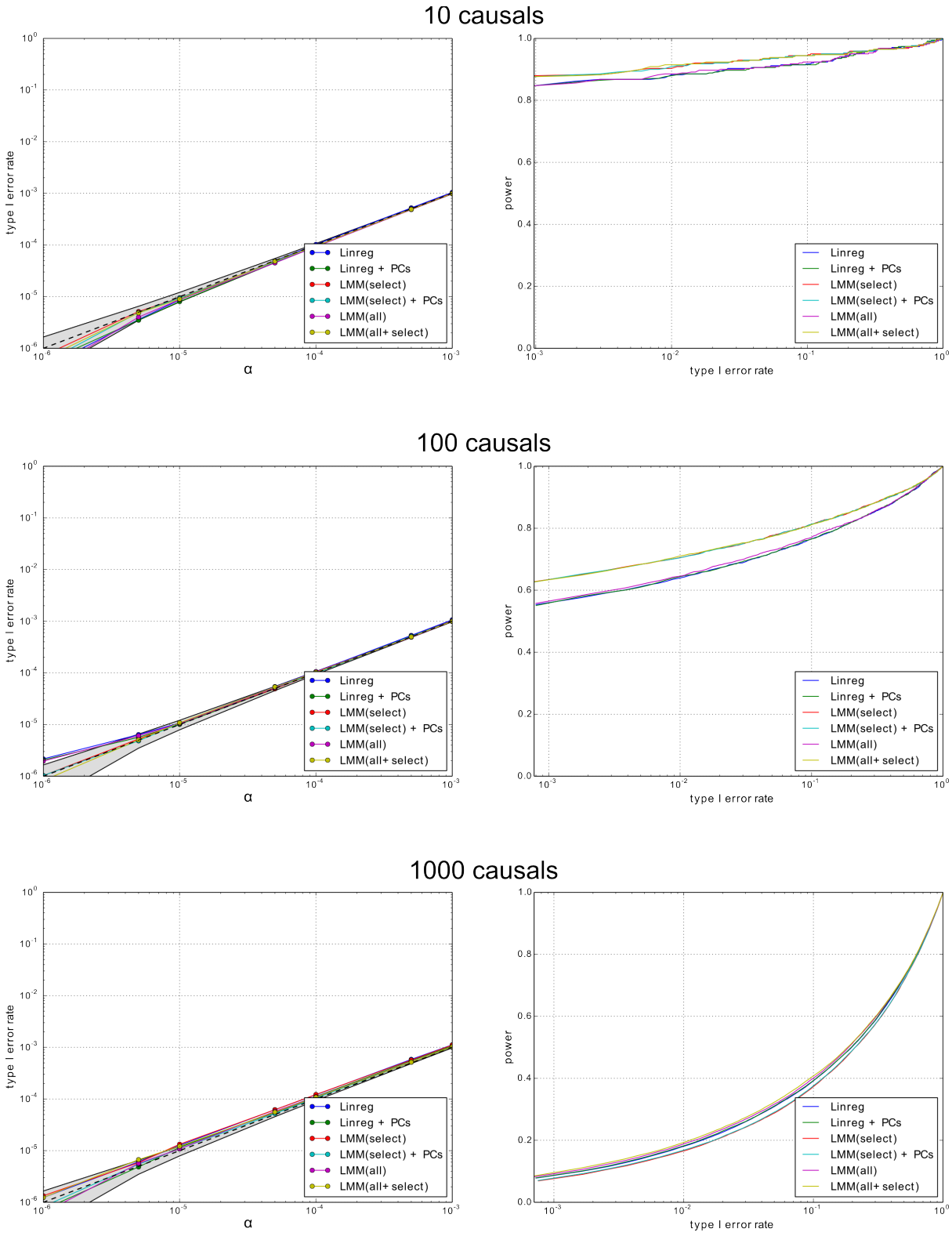
Supplementary Figure 4: Empirical type I error rate and power for family relatedness but no population structure with purely synthetic data. Each point represents the empirical type I error rate or power across 90 data sets with different degrees of signal (narrow-sense heritability) and family relatedness.



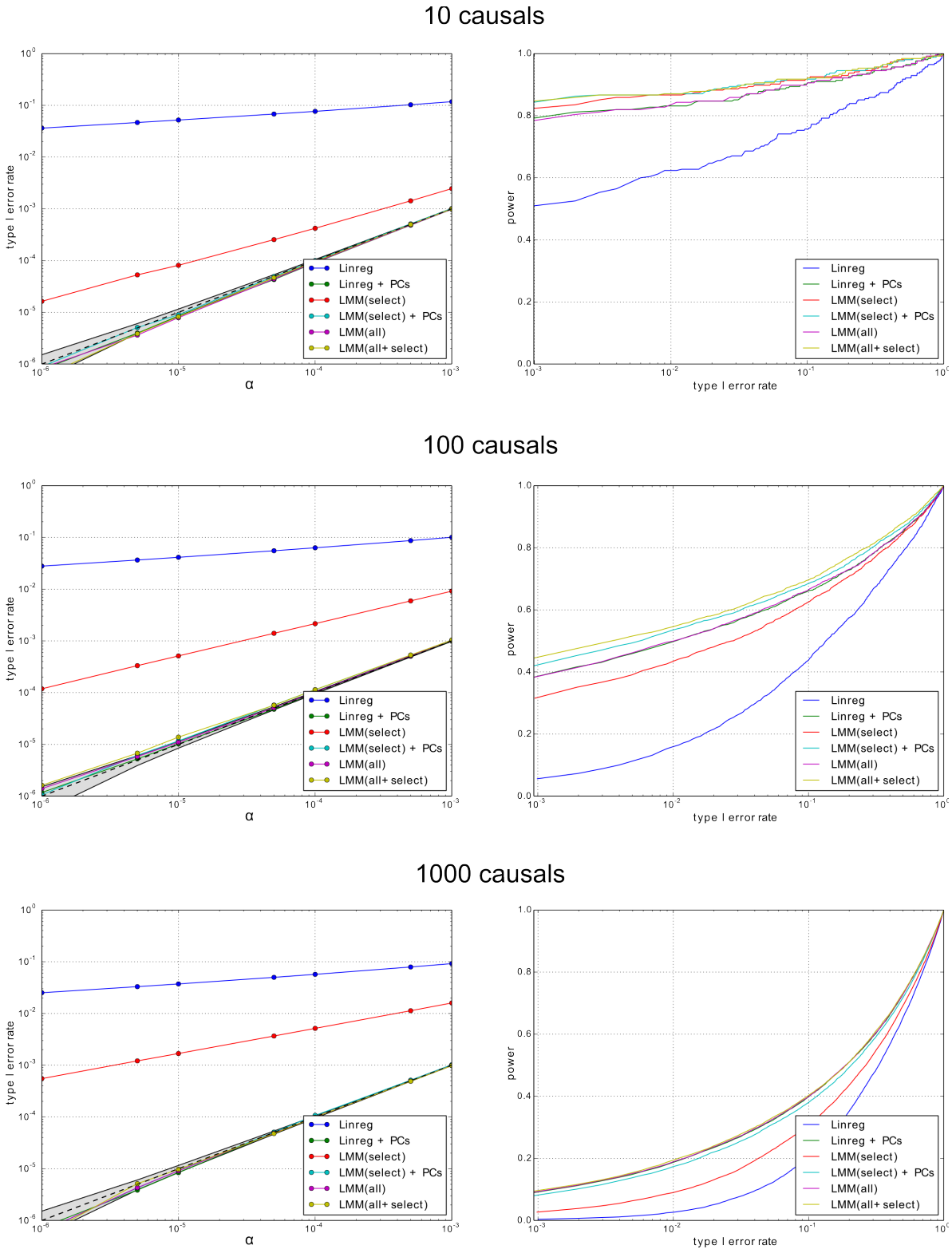
Supplementary Figure 5: Empirical type I error rate and power for family relatedness but no population structure, with and without a direct correlation between confounding structure and the phenotype. Each point represents the empirical type I error rate or power across 450 data sets with varying numbers of causal SNPs and with different numbers of causal SNPs and different degrees of signal (narrow-sense heritability) and family relatedness. The plots labeled “no direct correlation” and “direct correlation” correspond to the generating processes in **Figures 3a and 3b**, respectively.



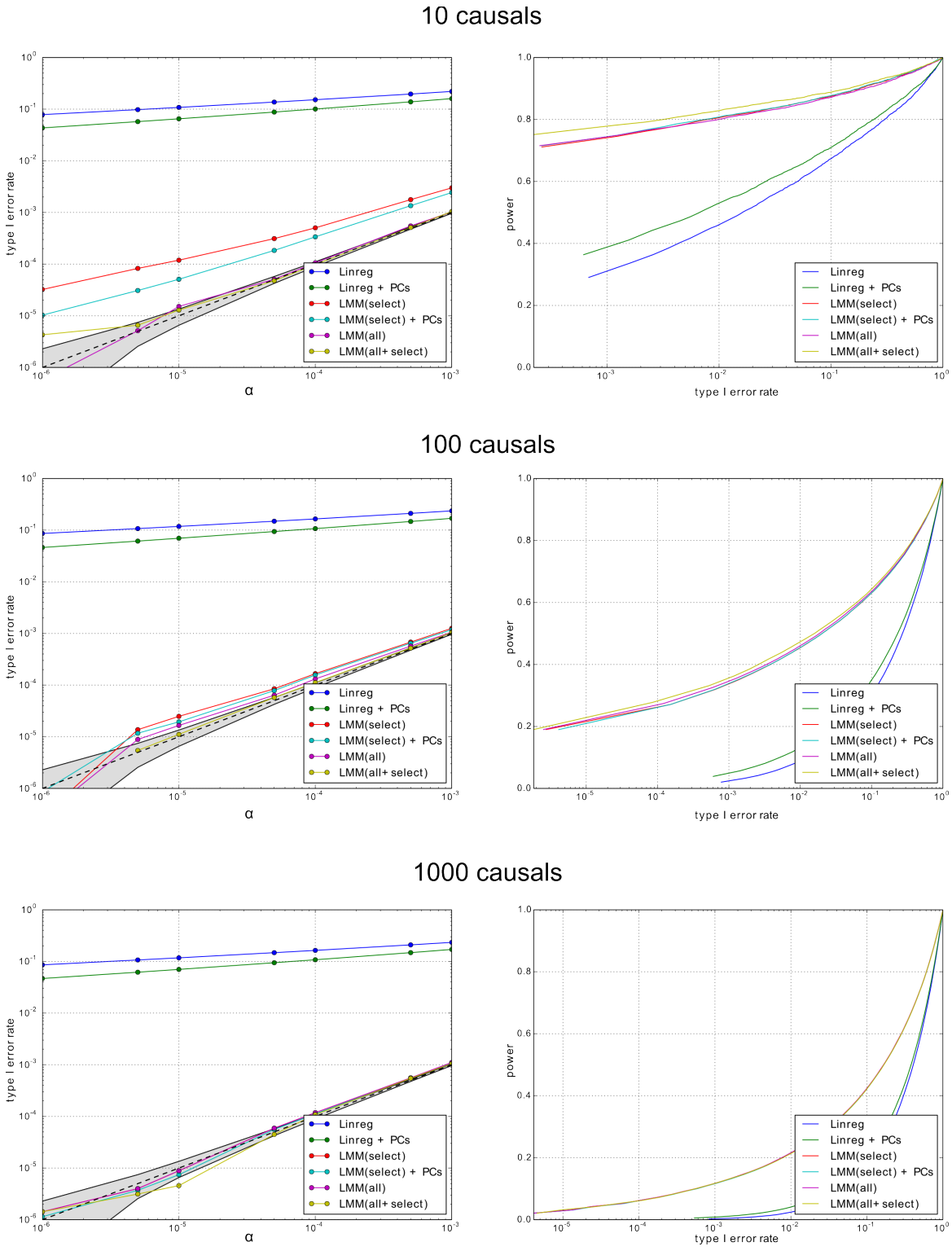
Supplementary Figure 6: Empirical type I error rate and power for both family relatedness and population structure with purely synthetic data. Each point represents the empirical type I error rate or power across 360 data sets with different degrees of signal (narrow-sense heritability), population structure, and family relatedness.



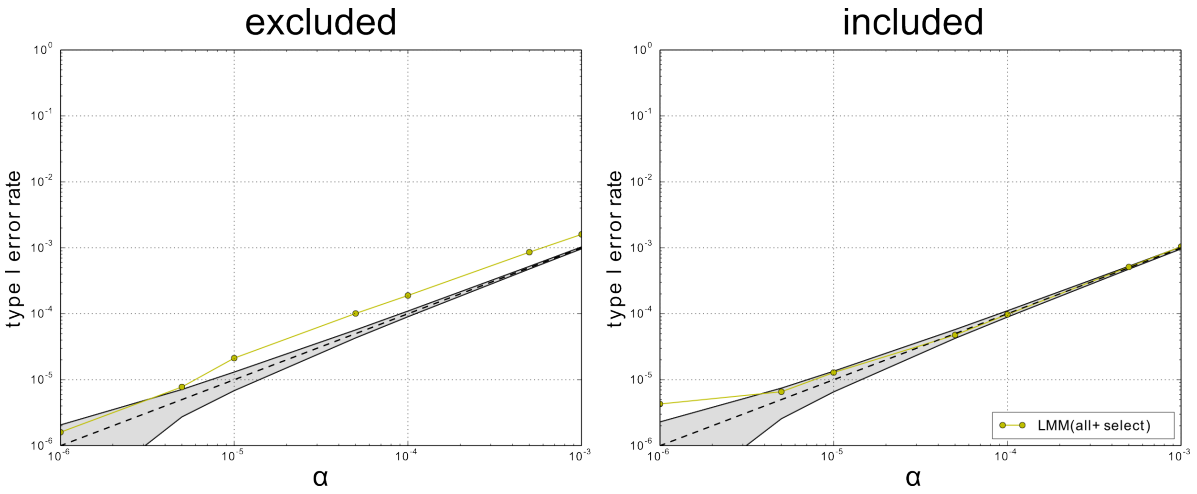
Supplementary Figure 7: Control of type I error and power for phenotypes synthetically generated from SNPs from the Finnish data. Each point represents empirical type I error rate or power across 400 synthetic phenotypes.



Supplementary Figure 8: Control of type I error and power for phenotypes synthetically generated from SNPs from the VAS data. Each point represents empirical type I error rate or power across 400 synthetic phenotypes.



Supplementary Figure 9: Control of type I error and power for phenotypes synthetically generated from SNPs from the Mouse data. Each point represents empirical type I error rate or power across 4,000 synthetic phenotypes.



Supplementary Figure 10: Control of type I error for phenotypes synthetically generated from SNPs from the Mouse data with 10 causal SNPs. Causal SNPs from one chromosome are either included or excluded from the GSM. Each point represents empirical type I error rate or power across 4,000 synthetic phenotypes.

Supplementary Table 1: Statistics for the analysis of real phenotypes from the Finnish and VAS data. The first column is the number of SNPs selected. The second column is the number of PCs after estimation. The remaining columns summarize GWAS performance based on the bronze standard (number of false positive and true positive loci under two different significance thresholds).

	# SNPs	# PCs	FP $5x10^{-8}$	TP $5x10^{-8}$	FP $5x10^{-7}$	TP $5x10^{-7}$
Finnish LDL						
Linreg	-	-	0	6	0	6
LMM(all)	328517	-	0	6	0	7
LMM(select)+PCs	32	13	0	6	0	7
LMM(all+select)	16	-	0	6	0	6
Finnish HDL						
Linreg	-	-	0	3	0	5
LMM(all)	328517	-	0	4	0	5
LMM(select)+PCs	16	13	0	3	0	4
LMM(all+select)	16	-	0	3	0	5
Finnish Triglyceride						
Linreg	-	-	0	2	0	2
LMM(all)	328517	-	0	2	0	2
LMM(select)+PCs	4	13	0	2	0	2
LMM(all+select)	2	-	0	2	0	2
AVS BMI						
Linreg	-	-	0	0	1	0
LMM(all)	720036	-	0	0	1	0
LMM(select)+PCs	720036	10	0	0	1	0
LMM(all+select)	4	-	0	0	2	0

A comparison of two methods for estimating PCs

We considered two methods for estimating PCs. The first method estimated PCs guided by the accuracy of phenotype prediction, similar to the approach of refs[1,2] for estimating PCs in combination with linear and logistic regression. In particular, the method tried increasing numbers of PCs associated with decreasing explained genetic variation (eigenvalues of the matrix of all SNPs), identifying the number of PCs that maximized the out-of-sample predictive accuracy on the phenotype. The method is called PCpheno (PC estimation based on phenotype) and, for the most complicated case where SNPs are also selected, was as follows:

1. Create random train-test partitions of the data samples.
2. For numPCs = 0
    a. For each partition
        i. Use the training data to compute univariate linear-regression association *P* values on each SNP using the PCs as covariates.
        ii. Order the SNPs by increasing *P* value.
        iii. For numSNPs in {0, 1, 2, 4, …, 8192, all} (the default values), use the first numSNPs of SNPs in the ordering as features for the LMM:
            1. Optimize the parameters of the LMM including δ using REML.
            2. Use the LMM to compute the predictive log likelihood of the test data.
3. Repeat step 2 with increasing numPCs until either (1) the predictive log likelihood of the test data maximized over numSNPs decreases twice in a row or (2) this log likelihood first increases and the decreases below the starting value. (These are the default values.)
4. Choose the value of numPCs and numSNPs that maximize jointly the sum over the partitions of the predictive log likelihood of the test data.

In the second method, the estimation of PCs was guided by the prediction accuracy of PCs on SNPs rather than the phenotype. In particular, we selected PCs by how well a corresponding probabilistic principal components analysis (PPCA) model (see next section and ref.[3]) maximized the predictive likelihood out-of-sample. The PPCA model captured statistical dependencies among the SNPs using a latent factor model, and included parameters that correspond to principal components. The method, called PCgeno (PC estimation based on genotype), was as follows:

1. Remove individuals that are closely related. Our default removes individuals until no two individuals have an estimated kinship coefficient from a GSM computed from all genome-wide SNPs of less than 0.1.
2. In the PPCA model, one dimension of the matrix of SNPs indexes multivariate samples, while the other indexes variables of the samples. To avoid problems due to the high dimensionality of the SNPs of each individual, treat the SNPs as samples and the individuals as variables.
3. Partition the samples into subsamples for cross-validation. Partition SNPs across chromosomes to reduce correlation between the partitions. When evaluating prediction

accuracy for a given subsample, the subsample is used for testing, while the other subsamples are used for training. Each train-and-test constitutes a fold.

4. For numPCs = 0
    a. For each fold
        i. Use the training data to compute maximum likelihood estimates for a PPCA model.
        ii. Compute the log likelihood of the test data according to this model.
5. Repeat step 4 with increasing numPCs until either (1) the sum over the folds of log likelihood decreases twice in a row or (2) this sum first increases and the decreases below the starting value.
6. Select the PPCA model that maximizes the sum over the folds of the log likelihood.
7. Project the individuals who were removed in step 1 onto the subspace defined by the optimal PPCA model (see next section).

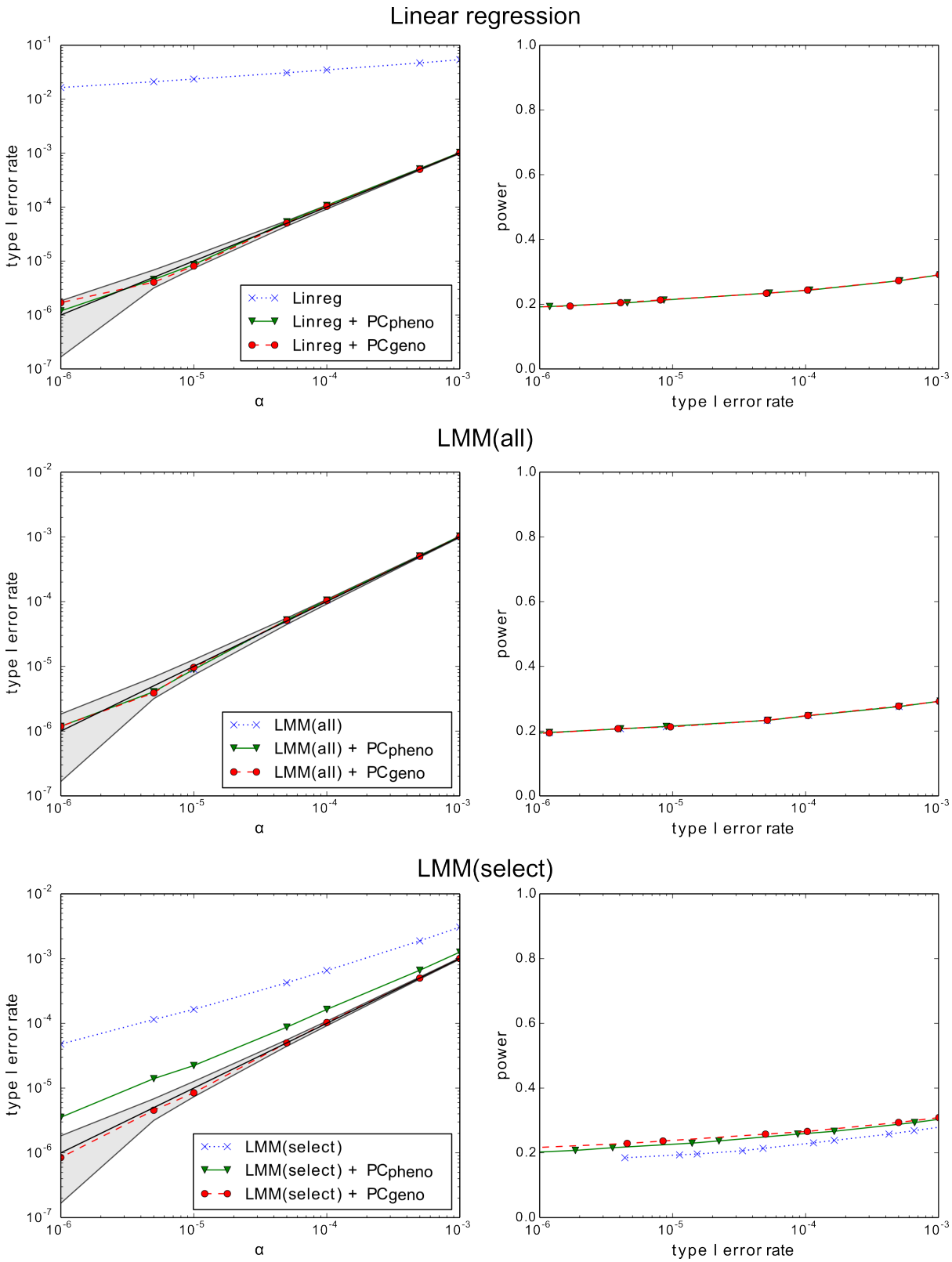
In this method, we used cross-validation rather than random train-test partitions. By using cross-validation and partitioning by chromosome (in step 3), independence between train and test sets was maximized.

Note that PCgeno determines PCs separately from the process of SNP selection. That is, PCgeno is used first to determine the PCs, and then given these PCs, a SNP selection algorithm is applied using the PCs as covariates. In contrast, PCpheno requires an integrated approach to PC estimation and SNP selection. The computation of PCs (in either PCpheno or PCgeno) has complexity $O(N^2M)$ in the simple case where all N PCs are computed.

On experiments with Linreg and LMM(SelectPheno) applied to SNPs having population structure but no family relatedness (generated as described in the main text), PCgeno outperformed PCpheno (**Supplemental Figure 11**).

The difference in performance between PCpheno and PCgeno on LMM(SelectPheno) can be understood in terms of the graphical model structure in **Figure 3a** of the main text. First, note that the portion of the graphical-model structure for SNP generation by the Balding-Nichols model—**Figure 3a** with y omitted—is identical to that for PPCA with one PC. Consequently, using one PC as a fixed effect is almost equivalent to conditioning on l, which would block paths from non-causal SNPs to y, leading to control of type I error. Now, when SNP selection was used, only weak paths from l to y remained, because most causal SNPs are conditioned on. Consequently, the PCpheno algorithm, which was guided by correlations between l and y, may have erroneously selected the wrong number of PCs (usually 0 in our experiments) and thus produced open paths from the non-causal SNPs to y. This hypothesis was validated by the results for Linreg. Namely, when we ran the PCpheno algorithm for Linreg (forcing numSNPs=0 in step 2c of the algorithm), there were stronger paths from l to y. As a result, the algorithm picked one PC across all data sets, resulting in good control of type I error.

PC estimation guided by the SNPs rather than the phenotype was not impacted by weak paths from l to y. Rather, the algorithm was able to recognize that a single latent variable could account for the correlations among the SNPs. Indeed, in our experiments, PCgeno always picked one PC.



Supplementary Figure 11: Empirical type I error rate and power for population structure but no family relatedness with purely synthetic data. Each point represents the empirical type I error rate or power across multiple data sets with different degrees of signal (narrow-sense heritability) and population structure.

The probabilistic principal components model

In the probabilistic principal components (PPC) model, each D dimensional sample is given by the linear model $x_i = \mu + u_i V^T + \epsilon_i$, where μ is a D dimensional mean vector, u_i is the k dimensional vector of principal components (PCs), the k times D matrix V^T are linear regression weights, also called loadings, and ϵ_i is a D dimensional noise vector. The principal components and the noise follow independent normal distributions:

$$u_i \sim N\left(0; \Lambda_k\right)$$
$$\epsilon_i \sim N(0; \sigma^2 I_D),$$

where Λ_k is a diagonal matrix and I is the identity matrix. The marginal likelihood follows from integrating out the PCs as well as the noise:

$$\prod_i \int N(x_i|\mu + u_i V^T; \sigma^2 I_D)\, N\left(u_i \middle| 0; \Lambda_k\right) du_i$$
$$= \prod_i N(x_i|\mu; V \Lambda V^T + \sigma^2 I_D).$$

The maximum likelihood values for the parameters $V, \Lambda, \mu, \sigma^2$ can be obtained from the singular value decomposition of the training data[3].

In the PC estimation method PCgeno, we estimate the parameters of the PPC model using a data set of non-closely related individuals and then project the remaining individuals onto the estimated PCs. To obtain the projection u_* for a new sample x_*, we maximize the likelihood of x_* with respect to u_* under the PPC model:

$$\log N(x_*|\mu + u_* V^T; \sigma^2 I_D) = \log N(x_* - \mu|u_* V^T; \sigma^2 I_D).$$

The likelihood of x_i is maximized by the least squares estimator for u_i:

$$\hat{u}_i = (x_i - \mu)(V^T V)^{-1} V^T$$

Thus, if the singular value decomposition $X = U \Lambda V^T$ is used to estimate V and Λ, then the columns of V are orthogonal, such that $V^T V = I_k$; and it follows that

$$\hat{u}_i = (x_i - \mu) V^T,$$
$$X = UV.$$

References

1. Novembre, J. & Stephens, M. Interpreting principal component analyses of spatial population genetic variation. *Nat. Genet.* **40,** 646–9 (2008).

2. S. Lee, F.A. Wright, and F. Z. Control of population stratification by correlation-selected principal components. *Biometrics* **67,** 967–974 (2011).

3. Tipping, M. E. & Bishop, C. M. Probabilistic Principal Component Analysis. *Analysis* 1–13 (1999).