UNIVERSITÉ
LIBRE
DE BRUXELLES

Computational Statistics

Report

2024/2025

Faculty of Sciences

Question 1

1. Problem Overview

It's given a data-generating process (DGP) defined by

$$Y_i \sim \text{Poisson}(\mu_i), \quad \mu_i = 5\cos(5x_i^2) + 2,$$

where $x_i \sim \mathcal{U}[0,1]$ for $i = 1, \ldots, n$ and $n = 100$. The objective is to model μ_i using a Generalized Linear Model (GLM) with a log link function:

$$g(\mu_i) = \log(\mu_i) = \theta_i = X_i\beta,$$

where X_i is a row of the design matrix formed by polynomial expansions of x_i up to degree m. The goal is to use AIC and BIC to determine the optimal polynomial degree $p \leq m$, then analyze what happens as $n \to \infty$, especially regarding BIC's consistency.

2. Results

e simulate data from the model $Y_i \sim \text{Poisson}(\mu_i)$, with $\mu_i = 5\cos(5x_i^2) + 2$ and $x_i \sim \text{Unif}[0,1]$, and fit GLMs using a log-link and polynomial basis of increasing degree p. AIC and BIC are used to select the optimal degree from nested submodels.

Sample Size n	Degree Selected by AIC (p_{AIC})	Degree Selected by BIC (p_{BIC})
100	7	5
1000	8	5
10000	8	8
20000	11	9
50000	11	11

BIC is *asymptotically consistent*: it selects the model that best approximates the DGP as $n \to \infty$. The empirical results support this, with selected degrees stabilizing as nn grows. As sample size increases, both AIC and BIC tend to select higher degrees, but BIC's choices reflect a convergence to an optimal balance between bias and variance.

Initially, AIC overfits (selecting larger p), while BIC selects more parsimonious models. However, as n becomes very large (e.g., n = 50,000), the selections by AIC and BIC begin to agree—both choosing degree 11. With enough data, the penalty terms become less dominant, and both criteria identify the same best-fitting model.

Overall, these results empirically support the theoretical consistency that BIC is **consistent**: as the sample size grows, it tends to select the model that best approximates the true data-generating process. They also suggest that even AIC may converge to the correct model as n grows, although it lacks formal consistency guarantees.

Figures

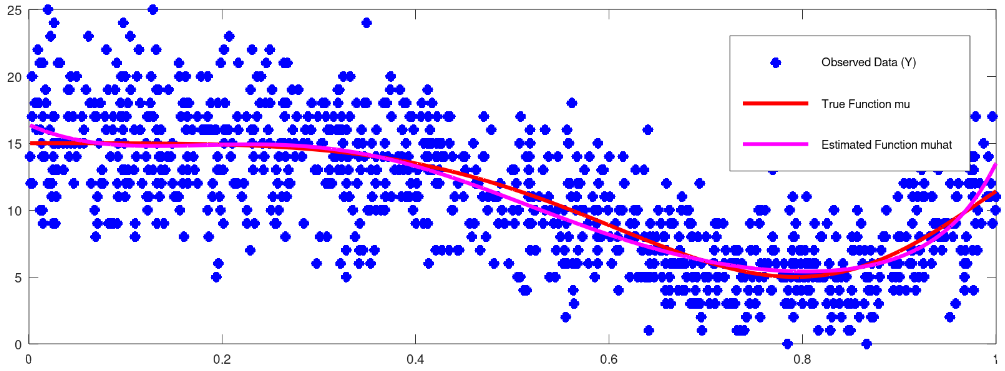


Figure 1: Observed data Y_i, true function μ_i, and estimated function $\hat{\mu}_i$ using the BIC-selected model (degree 5) for $n = 100$.

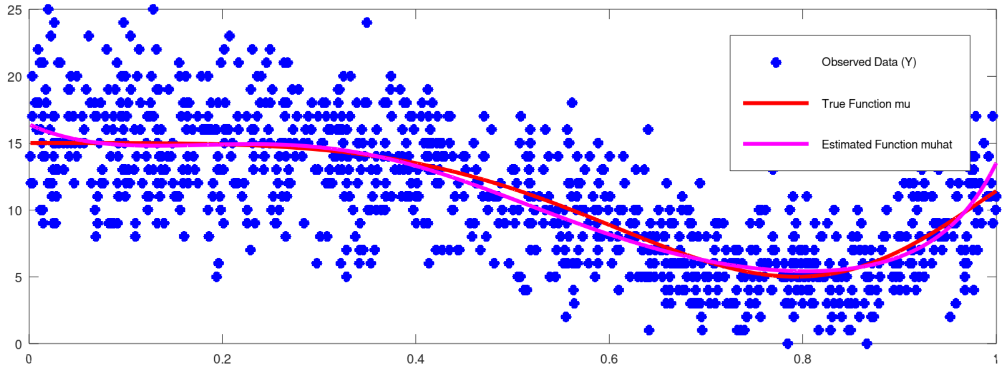


Figure 2: Observed data Y_i, true function μ_i, and estimated function $\hat{\mu}_i$ using the BIC-selected model (degree 5) for $n = 1000$.

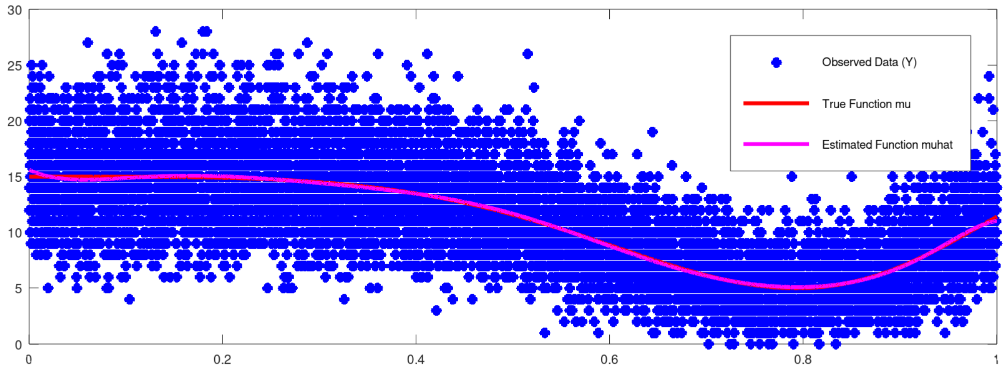


Figure 3: Observed data Y_i, true function μ_i, and estimated function $\hat{\mu}_i$ using the BIC-selected model (degree 8) for $n = 10000$.

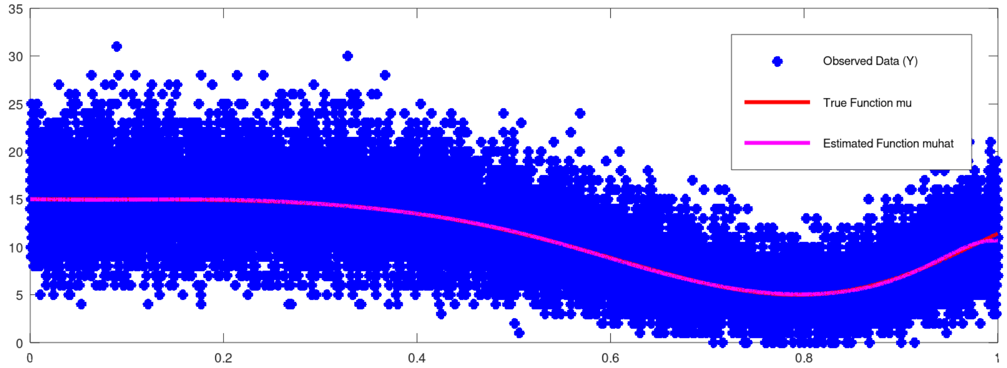


Figure 4: Observed data Y_i, true function μ_i, and estimated function $\hat{\mu}_i$ using the BIC-selected model (degree 9) for $n = 20000$.

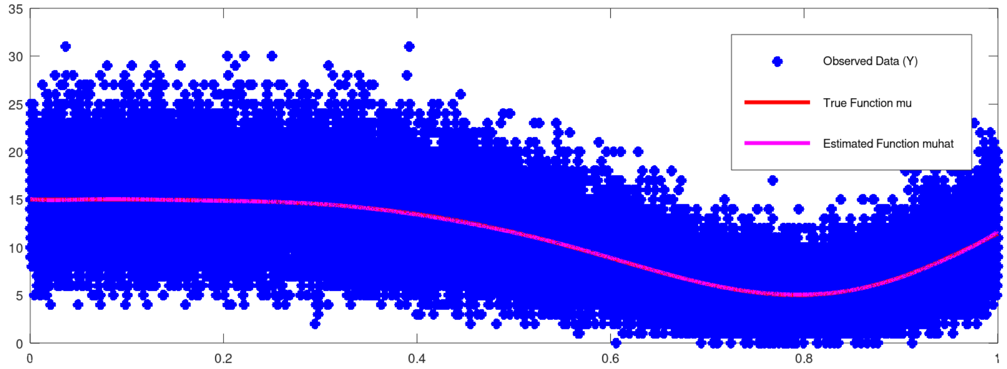


Figure 5: Observed data Y_i, true function μ_i, and estimated function $\hat{\mu}_i$ using the BIC-selected model (degree 11) for $n = 50000$.

Question 2

1. Problem Overview

I consider a simulation-based evaluation of model selection criteria in linear regression, where the true data-generating process (DGP) is a 7th-degree polynomial defined by its roots:

$$x \in \{-0.1000,\ 0.1555,\ 0.3143,\ 0.5469,\ 0.6903,\ 0.8730,\ 1.1\}.$$

This gives rise to a response variable:

$$Y_i = \beta_0 + \beta_1 x_i + \cdots + \beta_m x_i^m + \sigma Z_i,$$

where $Z_i \sim \mathcal{N}(0,1)$ and the DGP is a polynomial of degree 7.

The simulation investigates how model selection criteria—AIC, BIC, Mallows' C_p, prediction error (PE), and sum of squared difference $\|\mu - \hat{\mu}\|^2$, behave under increasing sample sizes:

$$n \in \{30,\ 100,\ 1000,\ 10000\}.$$

The focus is on how the curves for PE and Kullback-Leibler divergence (KL) change shape as n increases, and on comparing the asymptotic properties of AIC and BIC.

2. Results

Behavior of PE and KL Curves

As the sample size n increases, both the prediction error curve PE(p) and the Kullback-Leibler divergence KL(p) become increasingly flat on their right-hand side—that is, for model sizes $p > p^*$, where p^* is the optimal degree (in this case, 7).

This flattening arises because larger models (with $p > 7$) include the true model as a special case. Hence, although they introduce additional coefficients, these contribute less and less to improving the fit. For large n, the variance of the extra, unnecessary coefficients in an overfitted model becomes very small. Therefore, including them does not significantly harm the predictive accuracy, causing the PE and related error metrics to plateau beyond the true model complexity.

This behavior has practical implications: the error surface becomes less informative for penalizing complexity, making it harder to distinguish the optimal model from slightly overfitted ones based solely on PE or KL at large n.

The conclusions drawn about the behavior of the PE/KL curves are supported by the Figures below.

- **For n=30:** PE and squared error curves are V-shaped with a minimum near $p = 7$; overfitting is penalized clearly. As n increases, this minimum flattens beyond $p > 7$, reducing penalty for extra terms.
- **For n=100:** The minimum at $p = 7$ becomes more pronounced. The right-hand side of the curve ($p > 7$) is starting to flatten compared to the $n = 30$ case.
- **For n=1000 and n=10000:** The phenomenon becomes extremely clear. The curves exhibit a sharp drop to a minimum at exactly $p = 7$ and then become almost perfectly flat for all $p > 7$.

Comparison of AIC and BIC

The simulation clearly illustrates the fundamental asymptotic difference between AIC and BIC.

- **AIC** is designed for predictive accuracy and is not consistent. Its penalty for complexity does not grow with n, meaning it can favor overfitted models even in large samples.
- **BIC** is consistent. Its penalty term includes a $\log(n)$ factor, which increasingly penalizes complexity as n grows, allowing it to identify the true model in the large-sample limit.

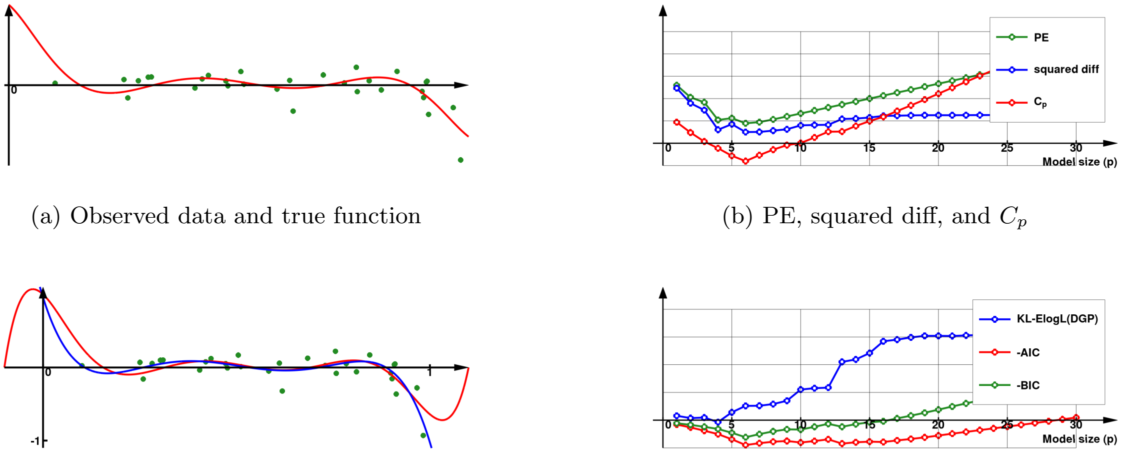
This theoretical difference is demonstrated in the plots:

1. **Evidence from Criterion Plots:** Mallows' C_p (the red curve), which is asymptotically equivalent to AIC, becomes completely flat for $p > 7$ in large samples. This shows that from AIC's perspective, there is no meaningful penalty for adding unnecessary predictors beyond the true model. In contrast, BIC's stronger penalty would ensure a clear minimum at $p = 7$.

2. **Evidence from Fitted Models:** The plots comparing the fitted polynomials (likely BIC in red vs. AIC in blue) visualize the consequence of this difference. The blue curve often appears more "wiggly" and complex, characteristic of an overfitted model selected by AIC. The red curve is smoother, representing the more parsimonious and true model selected by a consistent criterion like BIC.

In summary, the simulation confirms that BIC is more reliable for identifying the correct model structure in large datasets, whereas AIC's indifference to slight overfitting is a direct consequence of its formulation.

Figures



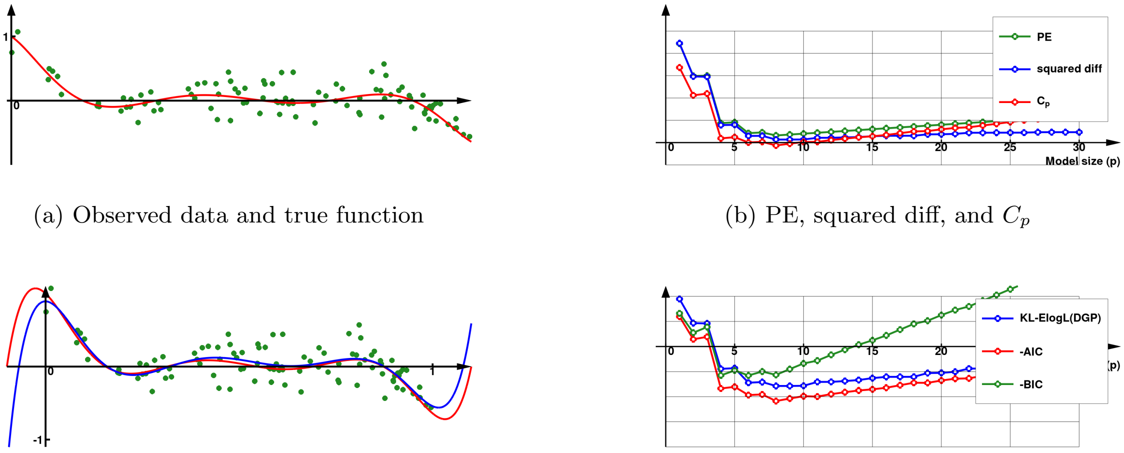
(a) Observed data and true function

(b) PE, squared diff, and C_p

(c) Best model fit on extended domain

(d) KL, AIC, and BIC vs. model size

Figure 6: Simulation results for $n = 30$.



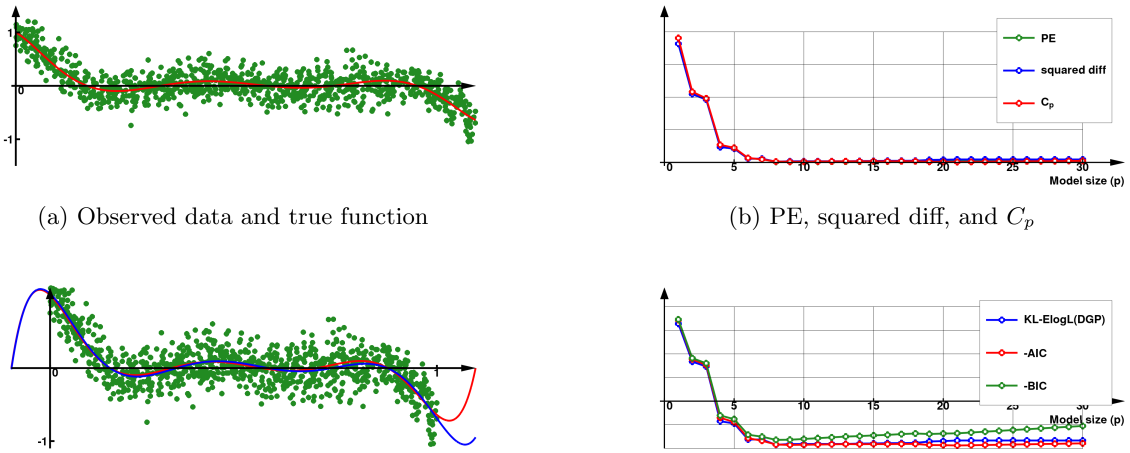
(a) Observed data and true function

(b) PE, squared diff, and C_p

(c) Best model fit on extended domain

(d) KL, AIC, and BIC vs. model size

Figure 7: Simulation results for $n = 100$.



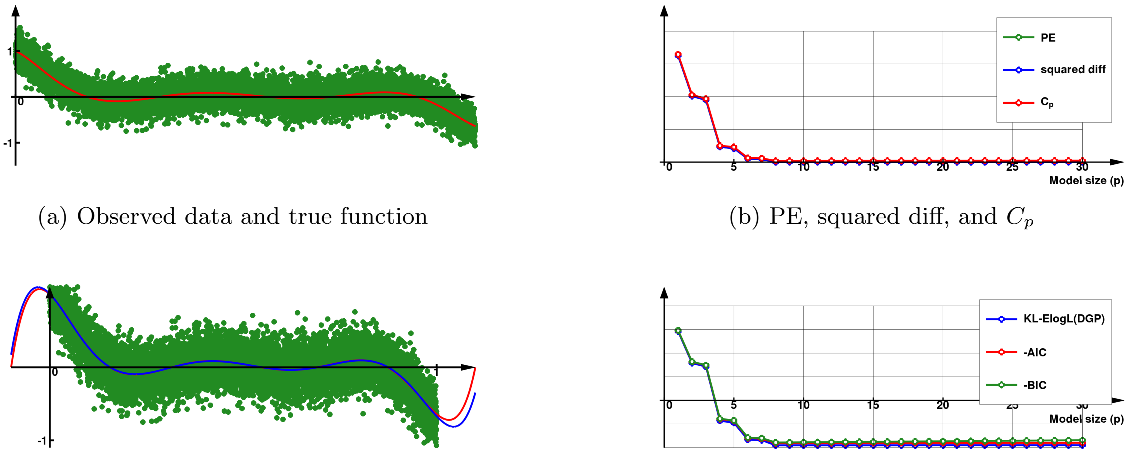
(a) Observed data and true function

(b) PE, squared diff, and C_p

(c) Best model fit on extended domain

(d) KL, AIC, and BIC vs. model size

Figure 8: Simulation results for $n = 1000$.



(a) Observed data and true function

(b) PE, squared diff, and C_p

(c) Best model fit on extended domain

(d) KL, AIC, and BIC vs. model size

Figure 9: Simulation results for $n = 10000$.

Question 3

1. Problem Overview

I consider a high-dimensional linear regression problem of the form

$$Y = X\beta + \sigma Z,$$

where β is a sparse coefficient vector, and $Z \sim \mathcal{N}(0, I_n)$ is standard normal noise. Sparsity in β is induced by sampling each entry independently to be zero with probability $1 - p$, and nonzero values are drawn from a double exponential (Laplace) distribution, which generates larger outliers than the normal errors.
The design matrix X is constructed using a band-diagonal structure via a kernel $K(\cdot)$, applied to a grid of inputs x_i and centers u_j, both sampled from uniform distributions. The kernel bandwidth h controls smoothness.
The simulation applies Lasso regression using the LARS algorithm to recover a sparse estimate $\hat{\beta}$, and evaluates the resulting fit and model selection via:

- Visual comparison of β vs. $\hat{\beta}$ and $\mu = X\beta$ vs. $\hat{\mu} = X\hat{\beta}$,
- Diagnostic plots of prediction error (PE), Mallows's C_p, and shrinkage paths,
- Analysis of the shrinkage path via $\log(\lambda)$,
- Verification of Karush-Kuhn-Tucker (KKT) conditions for the selected model.

2. Results

Comparison of True and Estimated Coefficients

The estimated coefficients $\hat{\beta}$ obtained by LARS are visually compared to the true sparse vector β. While the true β has few large entries and many exact zeros (by construction), the estimated $\hat{\beta}$ contains many small, nonzero values spread across the index range. This reflects the fact that Lasso tends to overshrink large coefficients and include small spurious entries.

In contrast, the corresponding comparison of the response vectors $\mu = X\beta$ and $\hat{\mu} = X\hat{\beta}$ shows much better agreement. The shrinkage in $\hat{\beta}$ still allows the estimator $\hat{\mu}$ to closely follow the structure of the true signal. This illustrates that, although Lasso may not accurately recover the support of β, it can still yield excellent predictions in terms of μ.

Effect of the Shrinkage Parameter λ

The shrinkage path is evaluated by plotting $\log(\lambda)$ against the model size κ. As κ increases, λ decreases monotonically, this reflects the behavior of LARS, which adds one variable at a time and decreases regularization step-by-step.
When $\log(\lambda) \approx 4$, corresponding to $\lambda \approx 54$, the model is extremely sparse, and nearly all coefficients are shrunk to zero. On the far right of the plot, when $\lambda \approx 0$, the model size reaches its maximum (e.g., $\kappa = 400$), and regularization essentially vanishes. This illustrates the bias-variance tradeoff: large λ leads to underfitting (high bias, low variance), while small λ allows overfitting (low bias, high variance).
In this experiment, the optimal model, based on C_p and prediction error, is associated with shrinkage levels around $\lambda \approx 0.56$ and $\lambda \approx 0.42$, respectively. These values correspond to moderate regularization and represent a point on the Lasso path where the model achieves a good tradeoff between sparsity and prediction accuracy.

KKT Conditions and Model Optimality

To verify that the Lasso solution is optimal, I check the Karush-Kuhn-Tucker (KKT) conditions. For active coefficients (those in S_{opt}), the KKT residuals $(X^\top r)_j$ should be exactly equal to $\lambda \cdot \text{sign}(\hat{\beta}_j)$. For inactive coefficients (those not in the model), they should lie within $[-\lambda, \lambda]$.

Plotting these values shows:

- The blue values (for $j \in S_{\text{opt}}$) correspond to the optimal nonzero coefficients under the Lasso penalty and lie sharply on the threshold value $[-\lambda, \lambda]$, as expected.
- The red values (for $j \notin S_{\text{opt}}$) mostly lie within the interval $[-\lambda, \lambda]$, although many lie very close to or directly on the threshold. This proximity weakens the strength of the KKT condition's satisfaction, suggesting borderline inclusion for some unselected variables.

This confirms that the solution satisfies the necessary optimality conditions at approximately $\lambda \approx 0.6$, reinforcing the validity, though not without caveat, of the LARS-based Lasso fit.

Figures

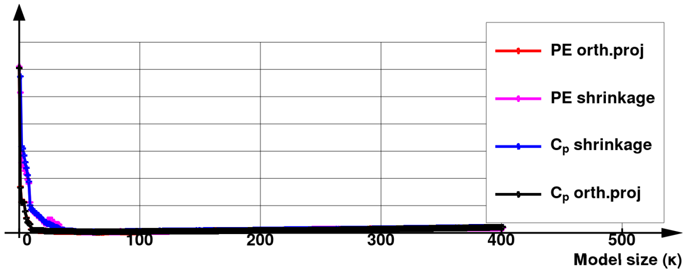


Figure 10: Prediction Error and C_p curves across model sizes.

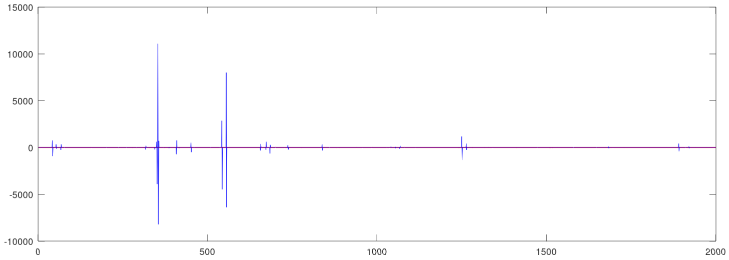


Figure 11: True β vs. estimated $\hat{\beta}$.

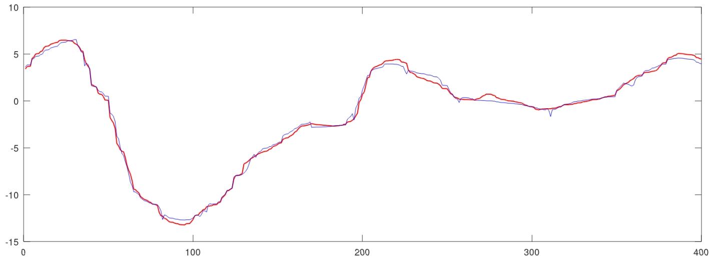


Figure 12: True signal μ and estimated $\hat{\mu}$.

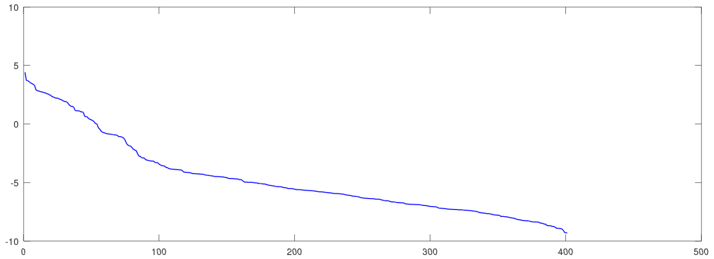


Figure 13: Shrinkage path: $\log(\lambda)$ vs. model size.

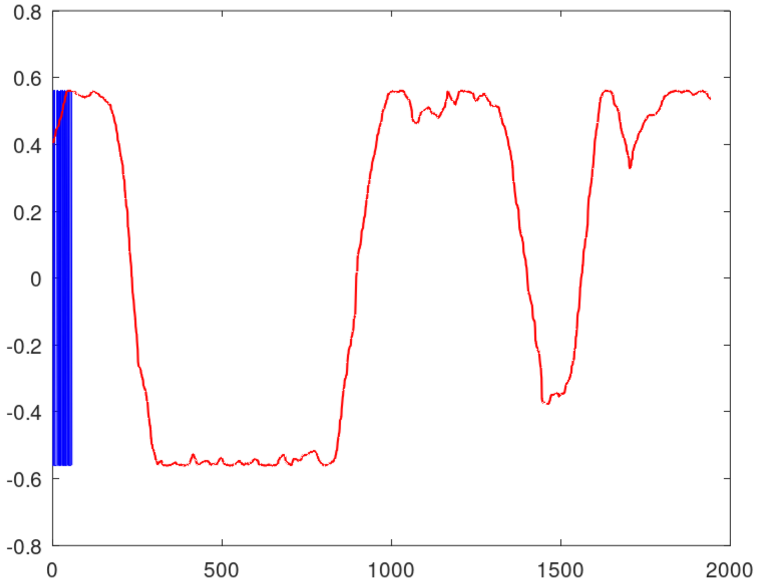


Figure 14: KKT residuals for selected and non-selected variables.

Question 4

1. Problem Overview

I consider the estimation of the population mean μ from a sum of independent Negative Binomial random variables $X_i \sim \text{NegBin}(p_i, r)$, where each variable counts the number of failures until the r-th success, with potentially different success probabilities p_i. The theoretical expressions for the mean and variance of the total sum $Y = \sum_{i=1}^{\kappa} X_i$ are:

$$\mu = r \sum_{i=1}^{\kappa} \left(\frac{1}{p_i} - 1 \right), \quad \sigma^2 = r \sum_{i=1}^{\kappa} \left(\frac{1 - p_i}{p_i^2} \right).$$

This setup is used to evaluate the performance of two types of bootstrap confidence intervals for the mean μ:

- **Basic bootstrap interval**, based on the quantiles of the bootstrap distribution of the mean.
- **Bootstrap-t interval**, based on quantiles of studentized bootstrap statistics.

The goal is to compare the empirical coverage probability and average width of the two interval types using repeated simulation.

2. Results

I simulated $ns = 1000$ independent samples for several sample sizes ($n \in \{20, 50, 100\}$). Each observation is the sum of $M = 3$ independent Negative Binomial variables with randomly drawn success probabilities. For each sample, I construct a 95% confidence interval using 10,000 bootstrap resamples. The empirical performance across the different sample sizes is summarized in the table below.

Table 1: Performance of 95% Bootstrap Confidence Intervals

Metric	**n = 20**	**n = 50**	**n = 100**
Basic Bootstrap Interval			
Coverage Probability	92.0%	93.1%	93.6%
Mean Interval Width	2.4685	1.6281	1.1622
Bootstrap-t Interval			
Coverage Probability	100%	100%	100%
Mean Interval Width	11.263	11.597	11.662

These results lead to the following conclusions:

- The **basic bootstrap interval**'s performance improves as the sample size increases. Its coverage probability gets closer to the nominal 95% level (from 92.0% to 93.6%), and as expected, the interval width narrows significantly. This demonstrates that it provides a reasonable trade-off between coverage and precision, becoming more reliable with more data.
- The **bootstrap-t interval** is consistently and extremely conservative across all sample sizes. It always achieves 100% coverage, far exceeding the nominal 95% level. This perfect coverage comes at a very high cost: the intervals are excessively wide (roughly 10 times wider than the basic bootstrap intervals) and do not narrow with increasing sample size, making it impractical despite theoretical appeal.

Overall, while the bootstrap-t method is often theoretically superior, this simulation shows it can be problematic in certain applications. The **basic bootstrap** method, despite its slight under-coverage in small samples, is clearly the more practical and balanced choice in this scenario, providing more informative (i.e., narrower) intervals that improve with more data.

Figures

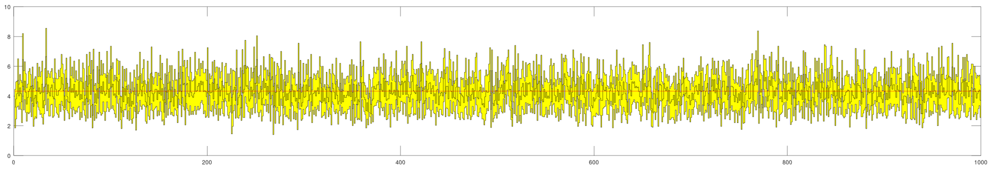


Figure 15: Basic bootstrap confidence intervals for μ, n=20.

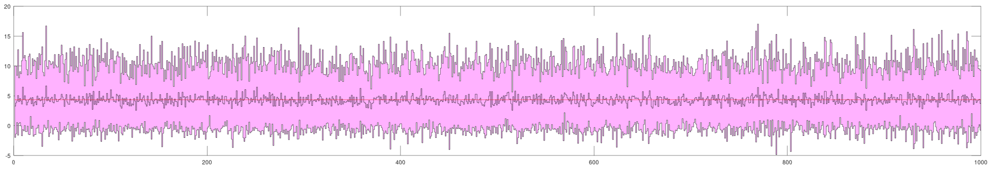


Figure 16: Bootstrap-t confidence intervals for μ, n=20.

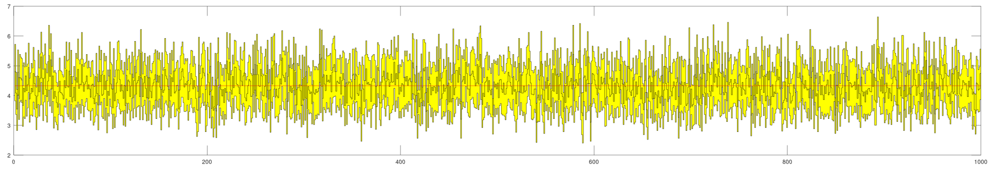


Figure 17: Basic bootstrap confidence intervals for μ, n=50.

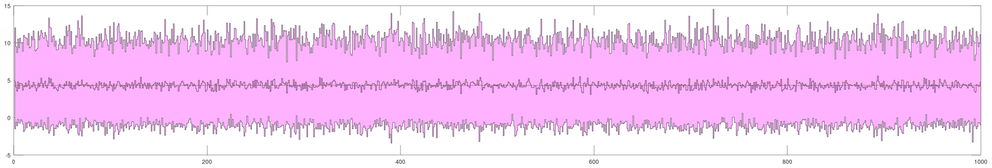


Figure 18: Bootstrap-t confidence intervals for μ, n=50.

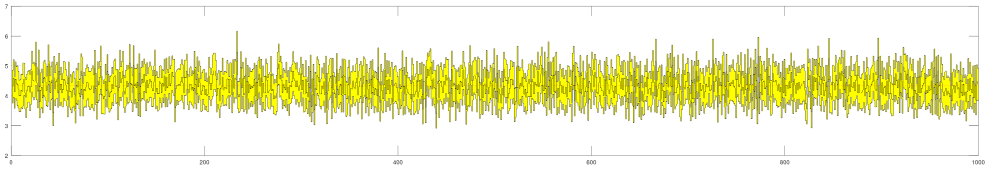


Figure 19: Basic bootstrap confidence intervals for μ, n=100.

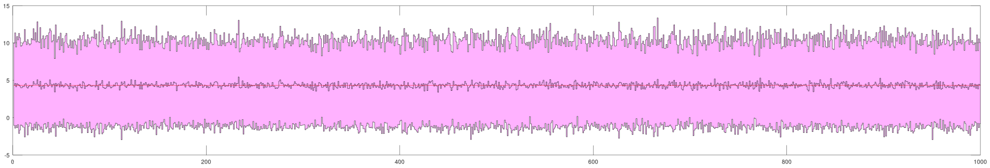


Figure 20: Bootstrap-t confidence intervals for μ, n=100.

Question 5

1. Problem Overview

It's given a target probability density function:

$$f_X(x) = K \cdot \exp\left(-\frac{1}{\sqrt{1-x^2}}\right), \quad x \in [-1, 1],$$

and a proposal density:

$$g_X(x) = L \cdot (1 - x^2), \quad x \in [-1, 1].$$

I aim to estimate $\mathrm{Var}(X)$ numerically using rejection sampling from $g_X(x)$.

2. Results

I seek a constant $M > 0$ such that:

$$f_X(x) \leq M \cdot g_X(x), \quad \forall x \in [-1, 1].$$

This holds if:

$$\frac{f(x)}{g(x)} = \frac{\exp\left(-\frac{1}{\sqrt{1-x^2}}\right)}{1-x^2} \leq \frac{ML}{K}.$$

I illustrate this in Figure 21.
I simulate a random variable $V \sim g_X(x)$ using the following algorithm:

```
a = 2; b = 2;
S = -log(rand(a+b,n));
U = sum(S(1:a,:)) ./ sum(S);
V = 2*U - 1;
```

This works because if $U \sim \mathrm{Beta}(a, b)$, then $V = 2U - 1$ has the desired shape $g_X(x) \propto 1 - x^2$ on $[-1, 1]$, due to the known connection between the Beta and semi-circular-like distributions.

Rejection Criterion

I accept a proposed value $V \sim g_X$ with probability:

$$U \leq \frac{f_X(V)}{M \cdot g_X(V)} = \frac{Kf(V)}{M \cdot Lg(V)}.$$

Since $f_X(x) = Kf(x)$ and $g_X(x) = Lg(x)$, the criterion simplifies to:

$$U \leq \frac{f(V)}{Mg(V)}.$$

I applied rejection sampling using the completed `randexpinvsqrt.m` function to generate pseudo-random samples from the target density:

$$f_X(x) = K \cdot \exp\left(-\frac{1}{\sqrt{1-x^2}}\right), \quad x \in [-1, 1].$$

Using 10^5 accepted samples drawn via the proposal distribution $g_X(x) \propto 1 - x^2$, I estimated the moments of the target distribution:

- Estimated mean: $\widehat{E}[X] \approx 0.00042$
- Estimated variance: $\widehat{\mathrm{Var}}(X) \approx 0.22401$

The near-zero mean is consistent with the symmetry of the density $f_X(x)$, which is even and centered around zero. The variance estimate reflects the spread induced by the rapidly decaying tail of the exponential factor in $f(x)$. This confirms the rejection sampling method's effectiveness for approximating expectations under complex, non-standard densities.

Figures

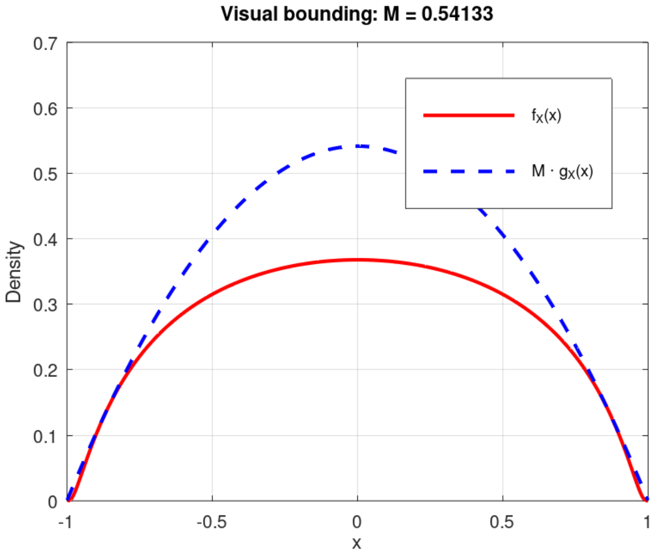


Figure 21: Plot of the target density $f(x)$, the proposal density $g(x)$, and the envelope $M \cdot g(x)$ over $[-1, 1]$. This shows that $f(x) \leq Mg(x)$ (assuming $K = L = 1$).

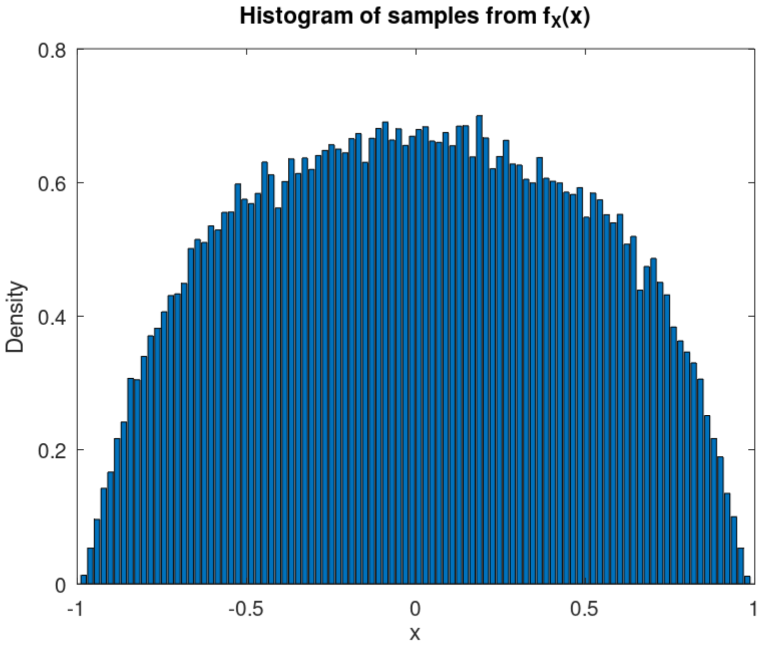


Figure 22: Histogram of samples drawn from $f_X(x)$ using rejection sampling. The distribution is symmetric and concentrated around zero, consistent with the shape of the target density.

Question 6

1. Problem Overview

In this exercise, I investigate and compare two Markov Chain Monte Carlo (MCMC) methods for sampling from a multivariate normal distribution: the **Gibbs sampler** and the **Metropolis-Hastings (MH)** algorithm. Both methods are used to simulate samples from a multivariate normal distribution $\mathcal{N}(0, \Sigma)$, where Σ is a known covariance matrix constructed from a random matrix A via $\Sigma = AA^\top$.

Although the multivariate normal distribution can be directly simulated using the transformation $Y = AZ$, where $Z \sim \mathcal{N}(0, I)$, this example illustrates the behavior of the two sampling algorithms in higher dimensions. For each dimension $d = 2, 3, \ldots, 8$, I generate 10,000 samples using each method and compare them to the true distribution.

In addition, I analyzed the proposal distribution used in the MH sampler and why it is valid in the MCMC context despite being unsuitable for direct rejection sampling of a normal distribution.

2. Results

Both Gibbs and Metropolis-Hastings samplers approximate the target distribution reasonably well for lower dimensions. The following behavior is observed as the dimension increases:

- **Gibbs sampler:** It maintains good performance across all tested dimensions. The estimated covariance matrix from the Gibbs samples closely matches the true covariance. The sampler mixes well, as each coordinate update is conditioned on the others, allowing for gradual but reliable convergence.

- **Metropolis-Hastings sampler:** The performance deteriorates with increasing dimension. The proposal distribution is uniform in a cube around the current state. As the dimension increases, the acceptance probability decreases due to the concentration of measure. Consequently, MH tends to explore the space less efficiently and suffers from slower convergence.

Figures below show a scatter plot of the dimensions considered $d = 2, 3, 4, 5, 6, 7, 8$. As dimension increases, the divergence between the true and MH samples becomes more pronounced, in particular whith $d = 8$.
The Metropolis-Hastings sampler uses a uniform random walk proposal:

$$y = x + \eta, \quad \eta \sim \text{Uniform}([-1, 1]^d).$$

This proposal is not suitable for rejection sampling of a normal distribution because:

- The normal density has unbounded support, while the uniform proposal is bounded.

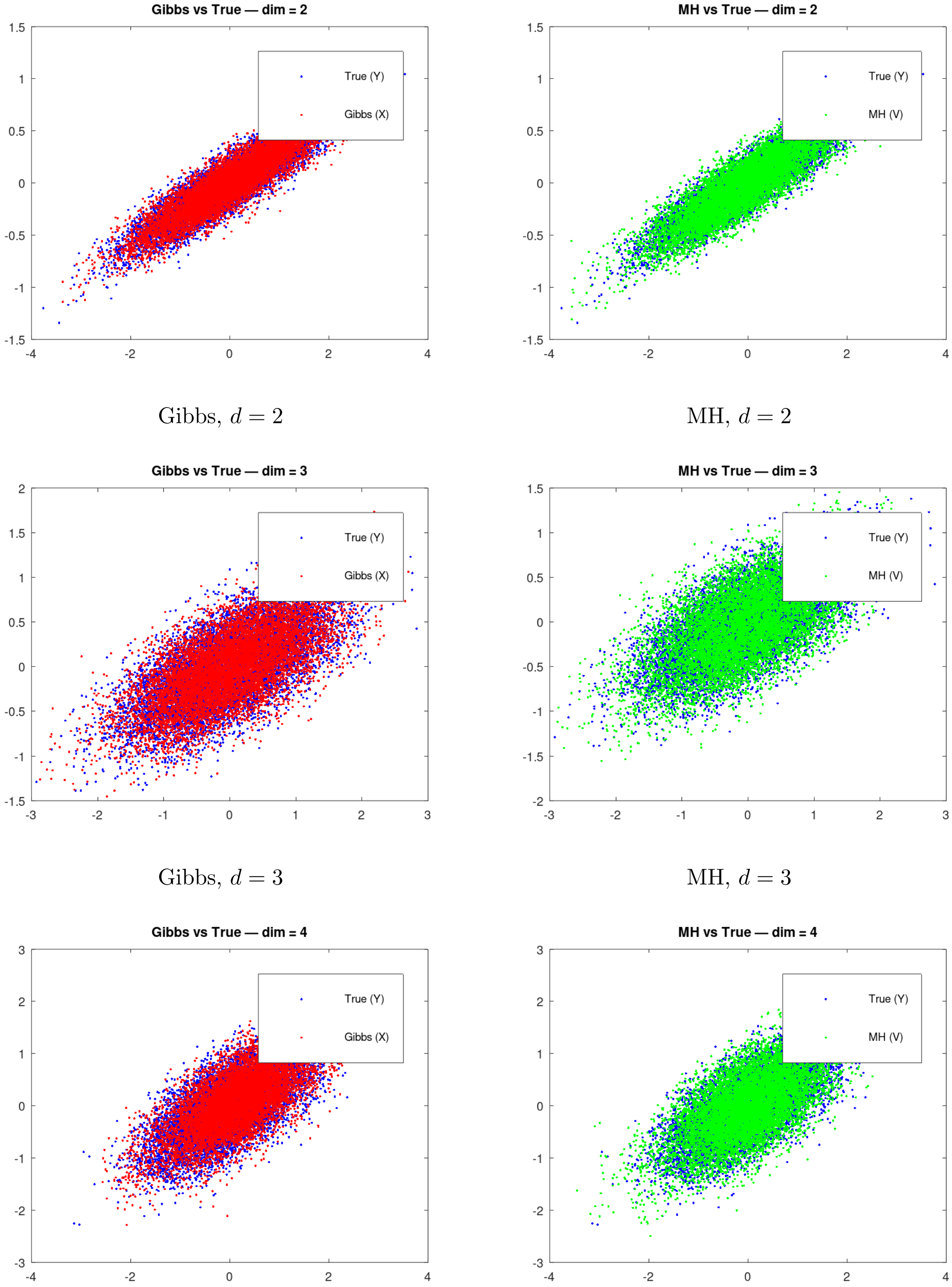
- There is no finite constant M such that the standard normal density $\phi(x) \leq M \cdot \text{Uniform}(x)$ for all $x \in R$.

However, this limitation is irrelevant in the Metropolis-Hastings context because MCMC allows rejection and correction of samples using the acceptance probability:

$$\alpha(x, y) = \min\left(1, \frac{\pi(y)}{\pi(x)}\right),$$

thus enabling valid sampling even with a crude proposal.

Figures



Gibbs, $d = 2$

MH, $d = 2$

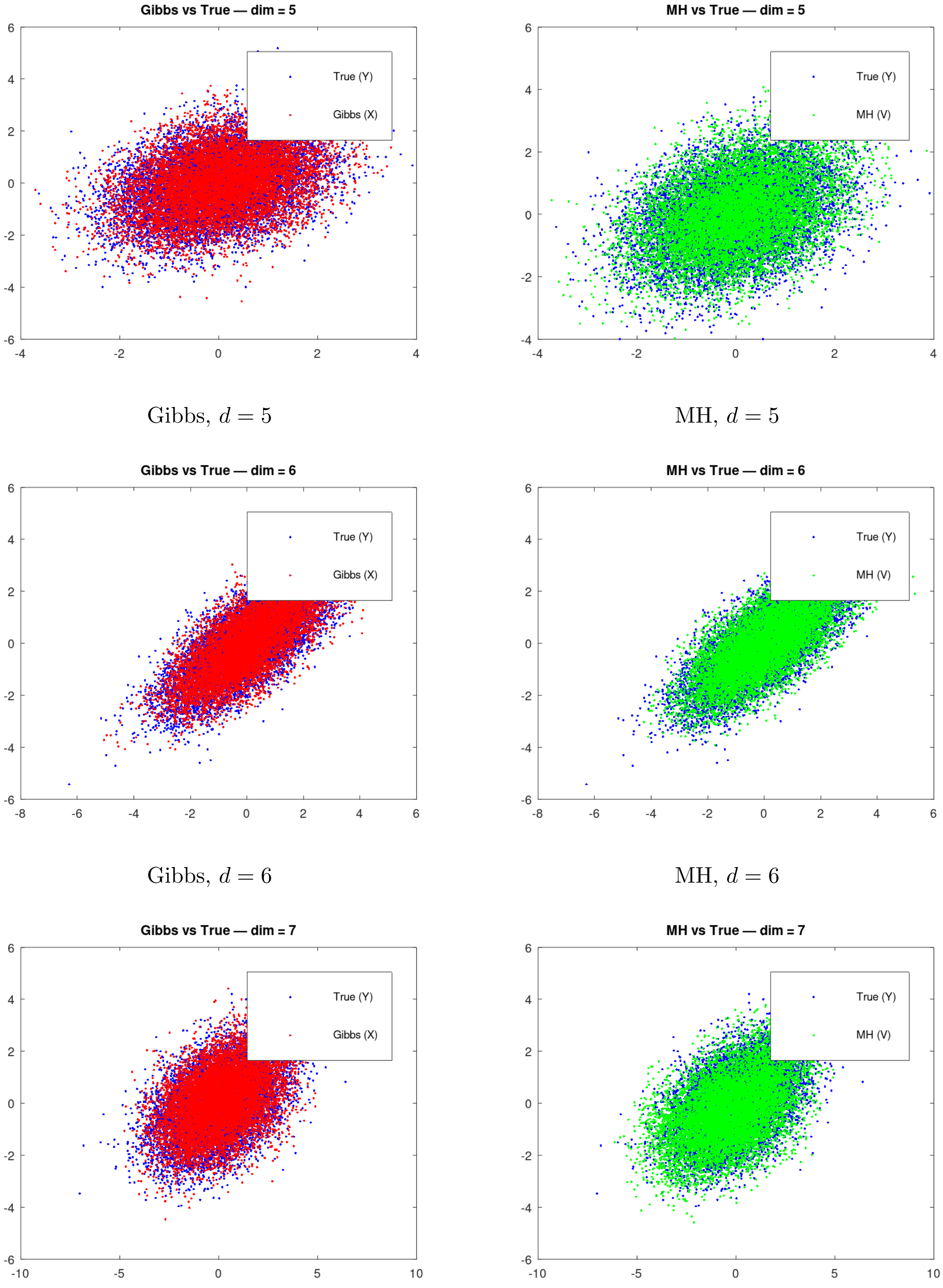
Gibbs, $d = 3$

MH, $d = 3$

Gibbs, $d = 4$

MH, $d = 4$

Figure 23: Comparison of Gibbs and Metropolis-Hastings samples with true samples across dimensions $d = 2, \dots, 4$.



Gibbs, $d = 5$

MH, $d = 5$

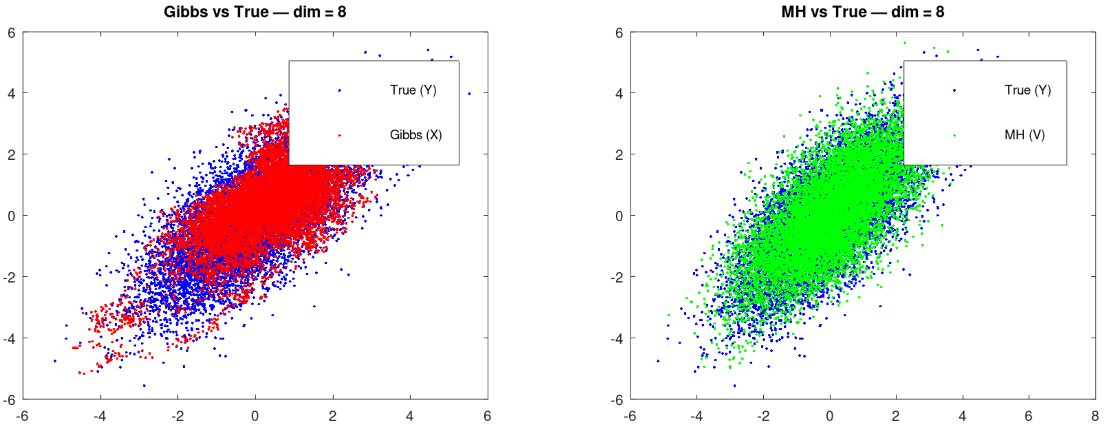
Gibbs, $d = 6$

MH, $d = 6$

Gibbs, $d = 7$

MH, $d = 7$

Figure 24: Comparison of Gibbs and Metropolis-Hastings samples with true samples across dimensions $d = 5, \ldots, 7$.



Gibbs, $d = 8$

MH, $d = 8$

Figure 25: Comparison of Gibbs and Metropolis-Hastings samples with true samples across dimensions $d = 8$.