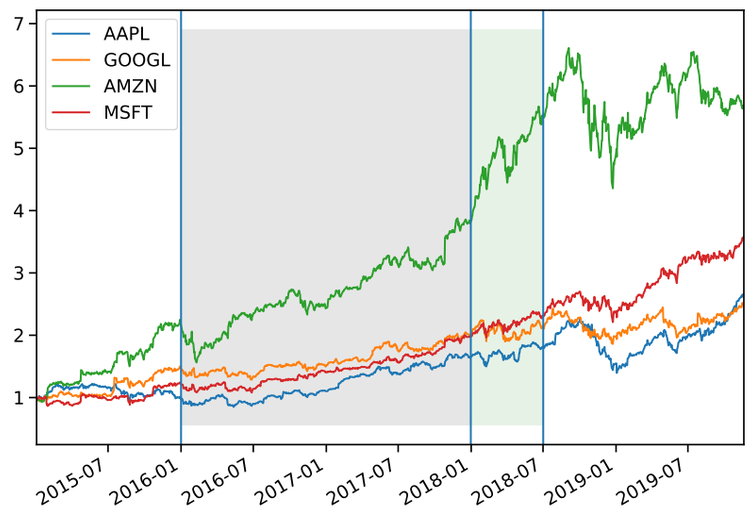
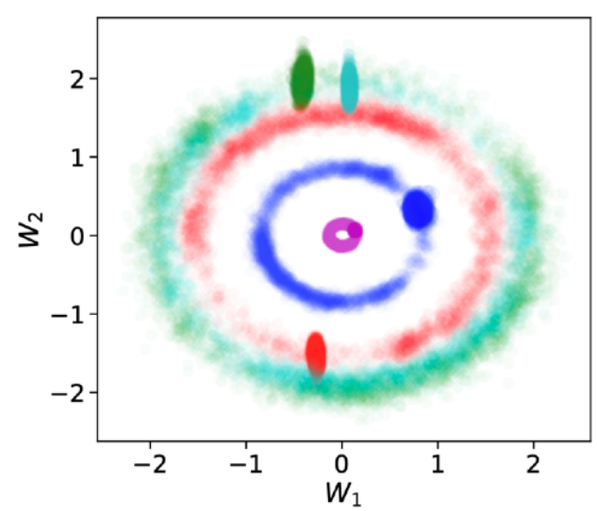
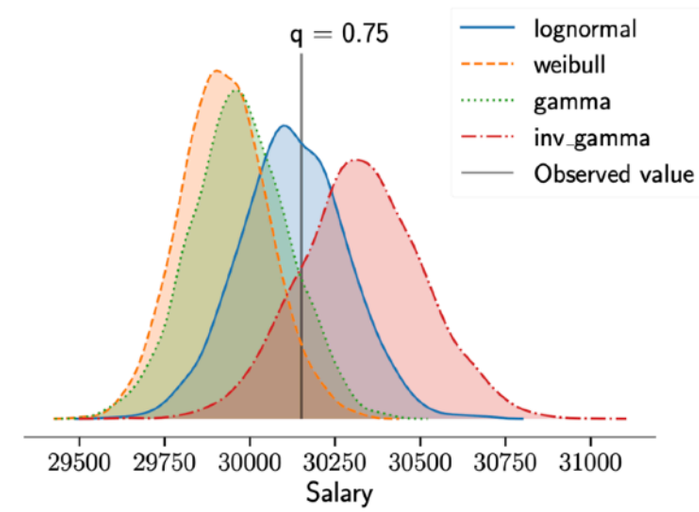
# Bayesian Machine Learning for Financial Modeling

Rajbir-Singh Nirwan

November 04, 2021




- Identification of structure in stock data using GPLVMs
- Solving problem of rotational symmetry in GPLVMs
- Proposing new likelihood for Quantile Matching Estimation
- Conclusion

# Gaussian Processes

Any finite collection of function values at $x_1, x_2, \ldots, x_N$ is jointly Gaussian distributed

$$p\left(f\left(x_1\right), f\left(x_2\right), \ldots, f\left(x_N\right)\right) = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{K})$$

# Gaussian Processes

Any finite collection of function values at $x_1, x_2, \ldots, x_N$ is jointly Gaussian distributed

$$p\left(f\left(x_1\right), f\left(x_2\right), \ldots, f\left(x_N\right)\right) = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{K}) = \mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \begin{pmatrix} k_{11} & k_{12} & \cdots & k_{1N} \\ k_{21} & k_{22} & \cdots & k_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ k_{N1} & k_{N2} & \cdots & k_{NN} \end{pmatrix}\right) \qquad k_{ij} = k(x_i, x_j)$$

# Gaussian Processes

Any finite collection of function values at $x_1, x_2, \ldots, x_N$ is jointly Gaussian distributed

$$p\left(f\left(x_1\right), f\left(x_2\right), \ldots, f\left(x_N\right)\right) = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{K}) = \mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \begin{pmatrix} k_{11} & k_{12} & \cdots & k_{1N} \\ k_{21} & k_{22} & \cdots & k_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ k_{N1} & k_{N2} & \cdots & k_{NN} \end{pmatrix}\right) \qquad k_{ij} = k(x_i, x_j)$$

Common Kernel Functions

$$k_{linear}(x, x') = xx'$$

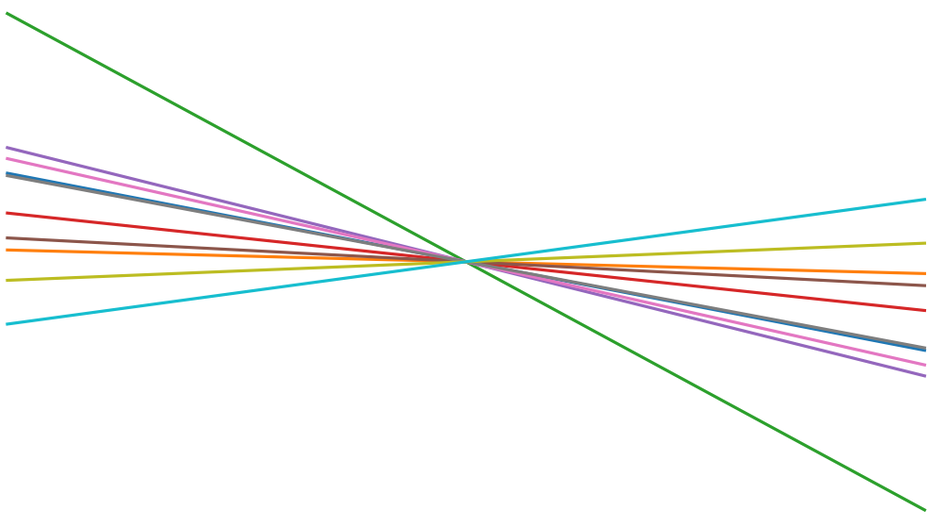
$$f \sim \mathbf{GP}(0, k(x_i, x_j))$$

$$k_{rbf}(x, x') = \exp\left(-\frac{1}{2\ell^2}(x - x')^2\right)$$

$$k_{ou}(x, x') = \exp\left(-\frac{1}{\ell}|x - x'|\right)$$

$$k_{mat32}(x, x') = \left(1 + \frac{\sqrt{3}\,|x - x'|}{\ell}\right)\exp\left(-\frac{\sqrt{3}\,|x - x'|}{\ell}\right)$$

$$k_{periodic}(x, x') = \exp\left(-\frac{2}{\ell^2}\sin^2(|x - x'|)\right)$$



# Gaussian Processes

Any finite collection of function values at $x_1, x_2, \ldots, x_N$ is jointly Gaussian distributed

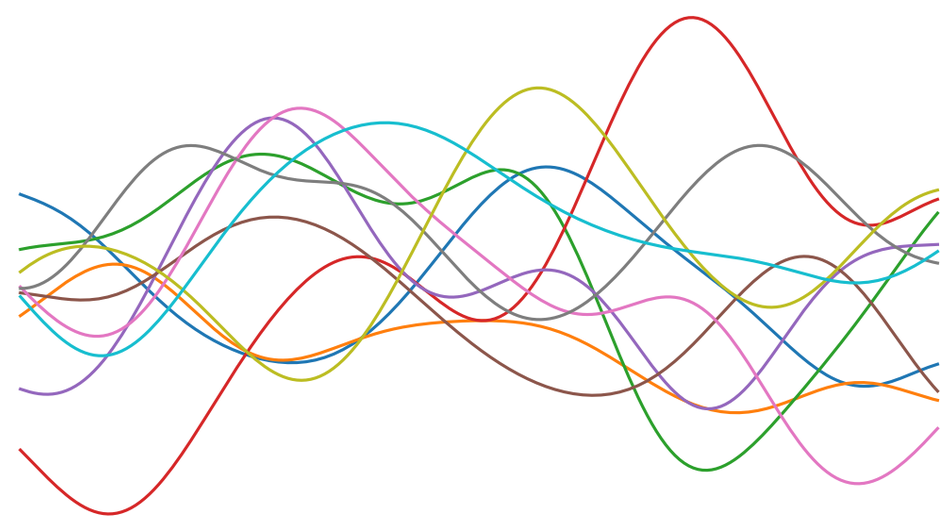
$$p\left(f\left(x_1\right), f\left(x_2\right), \ldots, f\left(x_N\right)\right) = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{K}) = \mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \begin{pmatrix} k_{11} & k_{12} & \cdots & k_{1N} \\ k_{21} & k_{22} & \cdots & k_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ k_{N1} & k_{N2} & \cdots & k_{NN} \end{pmatrix}\right) \qquad k_{ij} = k(x_i, x_j)$$

Common Kernel Functions

$$k_{linear}(x, x') = xx'$$

$$f \sim \mathbf{GP}(0, k(x_i, x_j))$$

$$k_{rbf}(x, x') = \exp\left(-\frac{1}{2\ell^2}(x - x')^2\right)$$



$$k_{ou}(x, x') = \exp\left(-\frac{1}{\ell}|x - x'|\right)$$

$$k_{mat32}(x, x') = \left(1 + \frac{\sqrt{3}\,|x - x'|}{\ell}\right)\exp\left(-\frac{\sqrt{3}\,|x - x'|}{\ell}\right)$$

$$k_{periodic}(x, x') = \exp\left(-\frac{2}{\ell^2}\sin^2(|x - x'|)\right)$$

# Gaussian Processes

Any finite collection of function values at $x_1, x_2, \ldots, x_N$ is jointly Gaussian distributed

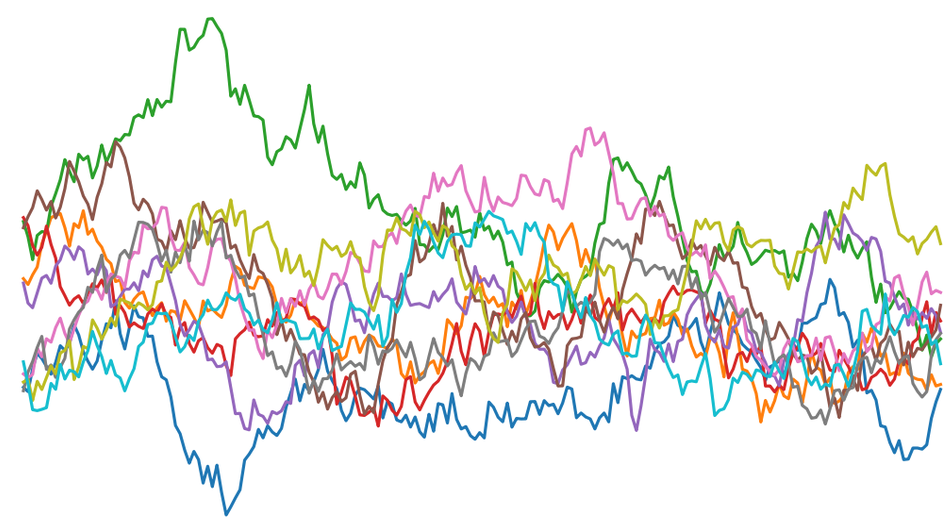
$$p\left(f\left(x_1\right), f\left(x_2\right), \ldots, f\left(x_N\right)\right) = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{K}) = \mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \begin{pmatrix} k_{11} & k_{12} & \cdots & k_{1N} \\ k_{21} & k_{22} & \cdots & k_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ k_{N1} & k_{N2} & \cdots & k_{NN} \end{pmatrix}\right) \qquad k_{ij} = k(x_i, x_j)$$

Common Kernel Functions

$$k_{linear}(x, x') = xx'$$

$$f \sim \mathbf{GP}(0, k(x_i, x_j))$$

$$k_{rbf}(x, x') = \exp\left(-\frac{1}{2\ell^2}(x - x')^2\right)$$



$$k_{ou}(x, x') = \exp\left(-\frac{1}{\ell}|x - x'|\right)$$

$$k_{mat32}(x, x') = \left(1 + \frac{\sqrt{3}\,|x - x'|}{\ell}\right)\exp\left(-\frac{\sqrt{3}\,|x - x'|}{\ell}\right)$$

$$k_{periodic}(x, x') = \exp\left(-\frac{2}{\ell^2}\sin^2(|x - x'|)\right)$$

# Gaussian Processes

Any finite collection of function values at $x_1, x_2, \ldots, x_N$ is jointly Gaussian distributed

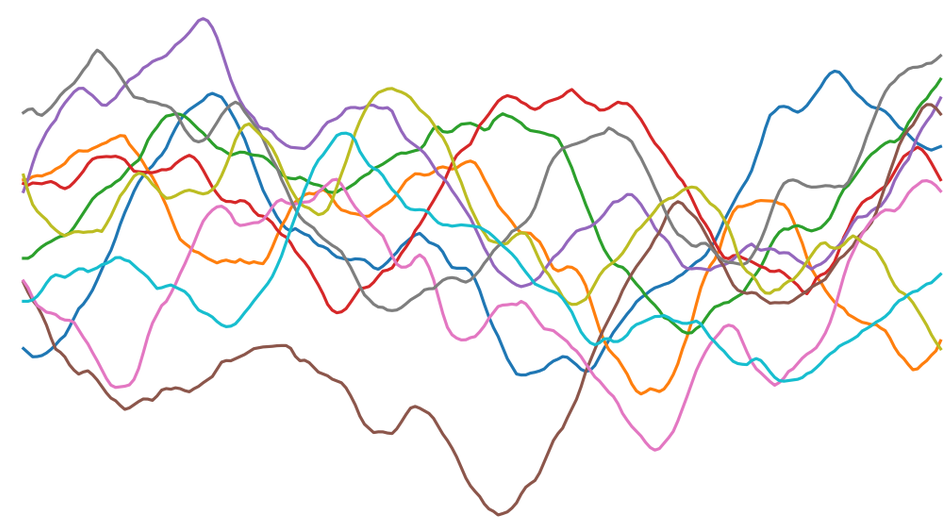
$$p\left(f\left(x_1\right), f\left(x_2\right), \ldots, f\left(x_N\right)\right) = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{K}) = \mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \begin{pmatrix} k_{11} & k_{12} & \cdots & k_{1N} \\ k_{21} & k_{22} & \cdots & k_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ k_{N1} & k_{N2} & \cdots & k_{NN} \end{pmatrix}\right) \qquad k_{ij} = k(x_i, x_j)$$

Common Kernel Functions

$$k_{linear}(x, x') = xx'$$

$$f \sim \mathbf{GP}(0, k(x_i, x_j))$$

$$k_{rbf}(x, x') = \exp\left(-\frac{1}{2\ell^2}(x - x')^2\right)$$



$$k_{ou}(x, x') = \exp\left(-\frac{1}{\ell}|x - x'|\right)$$

$$k_{mat32}(x, x') = \left(1 + \frac{\sqrt{3}\,|x - x'|}{\ell}\right)\exp\left(-\frac{\sqrt{3}\,|x - x'|}{\ell}\right)$$

$$k_{periodic}(x, x') = \exp\left(-\frac{2}{\ell^2}\sin^2(|x - x'|)\right)$$

# Gaussian Processes

Any finite collection of function values at $x_1, x_2, \ldots, x_N$ is jointly Gaussian distributed

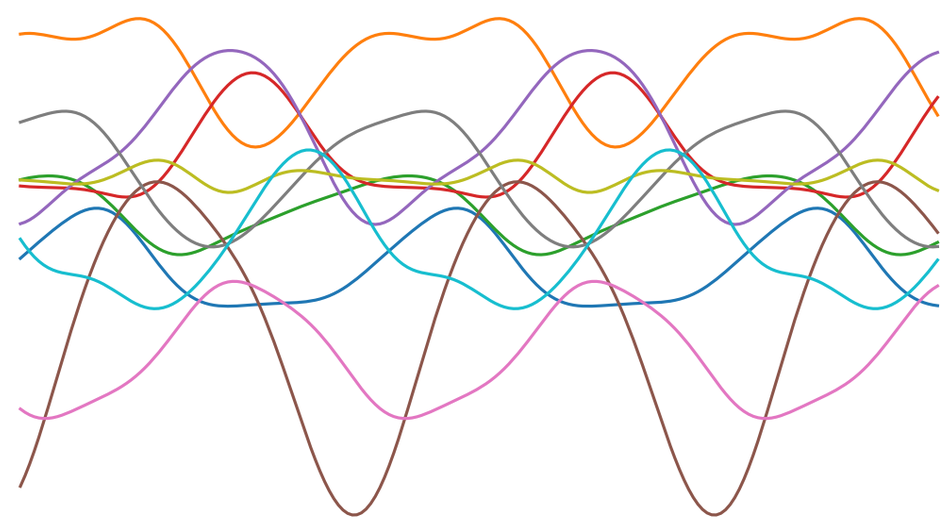
$$p\left(f\left(x_1\right), f\left(x_2\right), \ldots, f\left(x_N\right)\right) = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{K}) = \mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \begin{pmatrix} k_{11} & k_{12} & \cdots & k_{1N} \\ k_{21} & k_{22} & \cdots & k_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ k_{N1} & k_{N2} & \cdots & k_{NN} \end{pmatrix}\right) \qquad k_{ij} = k(x_i, x_j)$$

Common Kernel Functions

$$k_{linear}(x, x') = xx'$$

$$f \sim \mathbf{GP}(0, k(x_i, x_j))$$

$$k_{rbf}(x, x') = \exp\left(-\frac{1}{2\ell^2}(x - x')^2\right)$$



$$k_{ou}(x, x') = \exp\left(-\frac{1}{\ell}|x - x'|\right)$$

$$k_{mat32}(x, x') = \left(1 + \frac{\sqrt{3}\,|x - x'|}{\ell}\right)\exp\left(-\frac{\sqrt{3}\,|x - x'|}{\ell}\right)$$

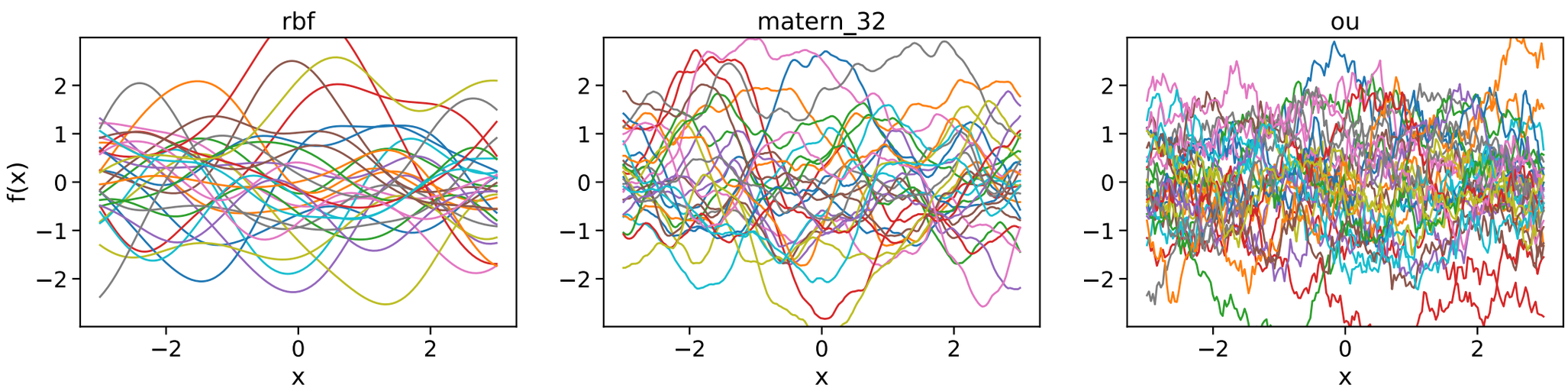
$$k_{periodic}(x, x') = \exp\left(-\frac{2}{\ell^2}\sin^2(|x - x'|)\right)$$

# Gaussian Processes

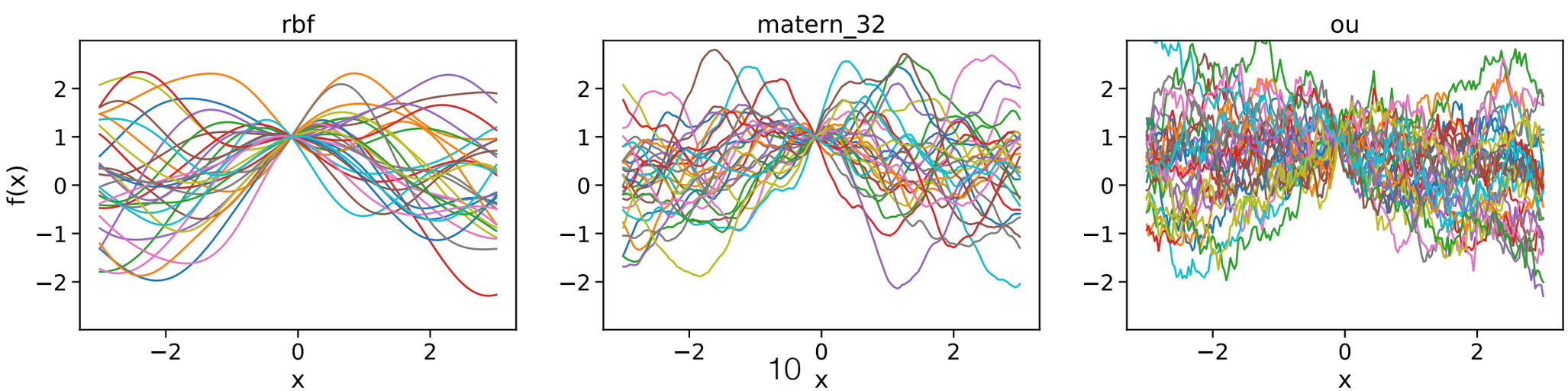
**Bayes Theorem**

$$p(f \mid \mathcal{D}) = \frac{p(\mathcal{D} \mid f)p(f)}{p(\mathcal{D})}$$

**Prior**




**Posterior**

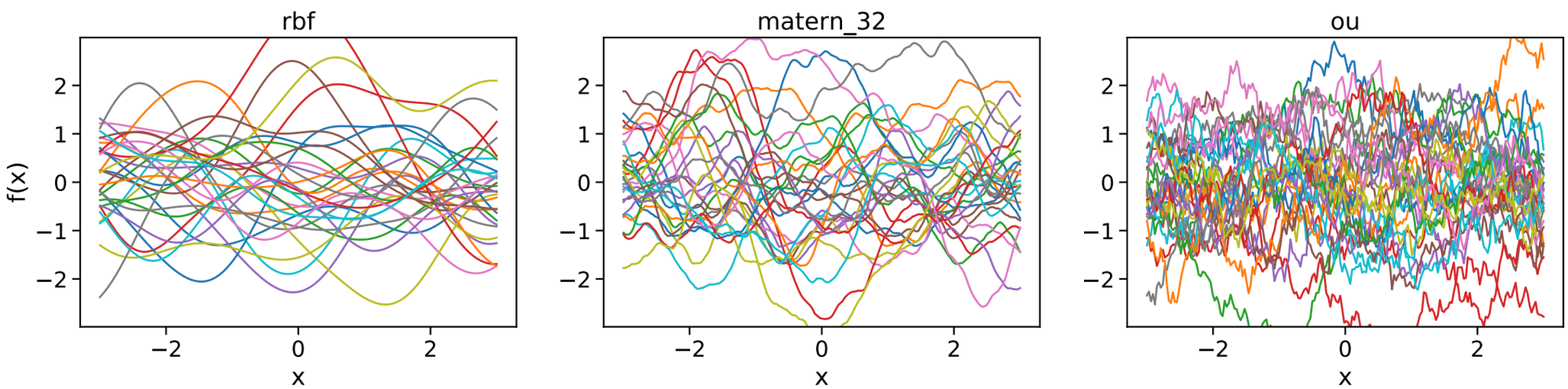


# Gaussian Processes

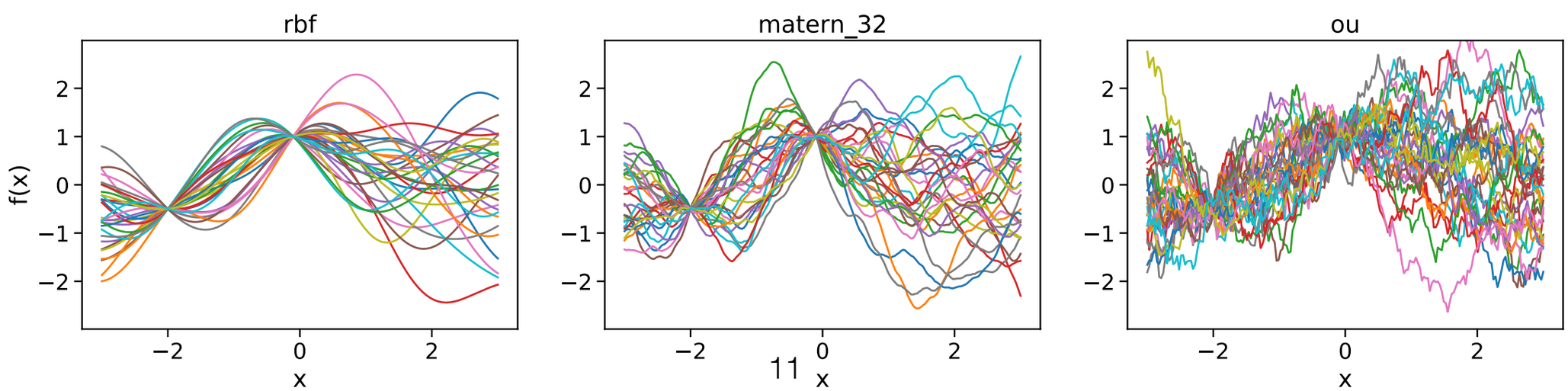
**Bayes Theorem**

$$p(f \mid \mathcal{D}) = \frac{p(\mathcal{D} \mid f)p(f)}{p(\mathcal{D})}$$

**Prior**




**Posterior**

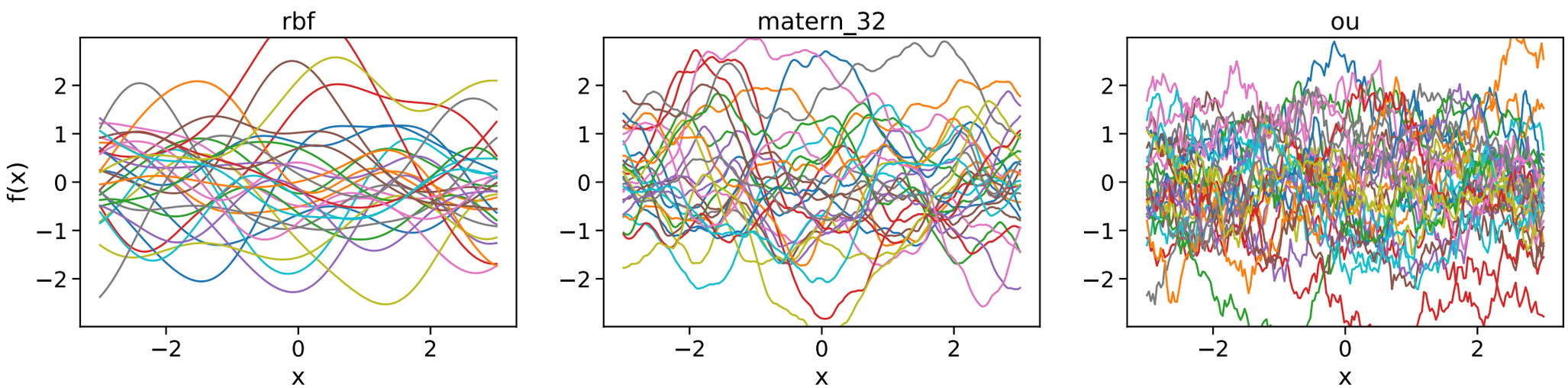


# Gaussian Processes

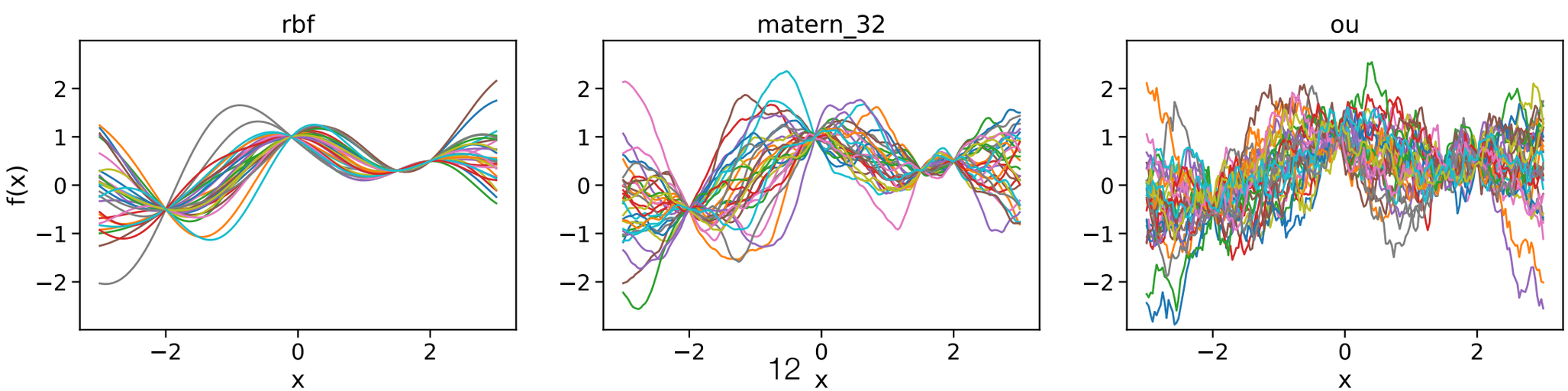
**Bayes Theorem**

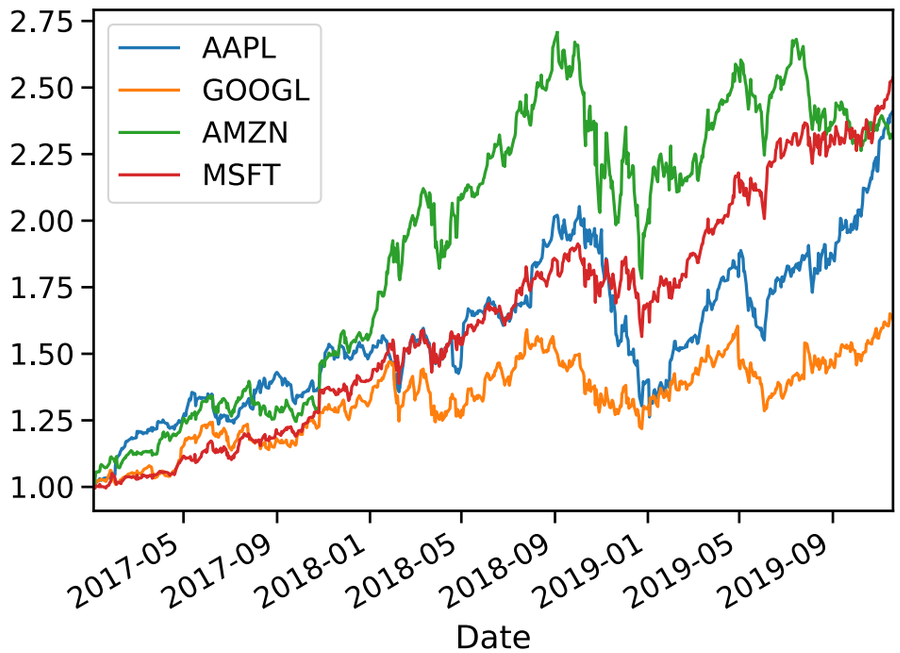
$$p(f \mid \mathcal{D}) = \frac{p(\mathcal{D} \mid f)p(f)}{p(\mathcal{D})}$$

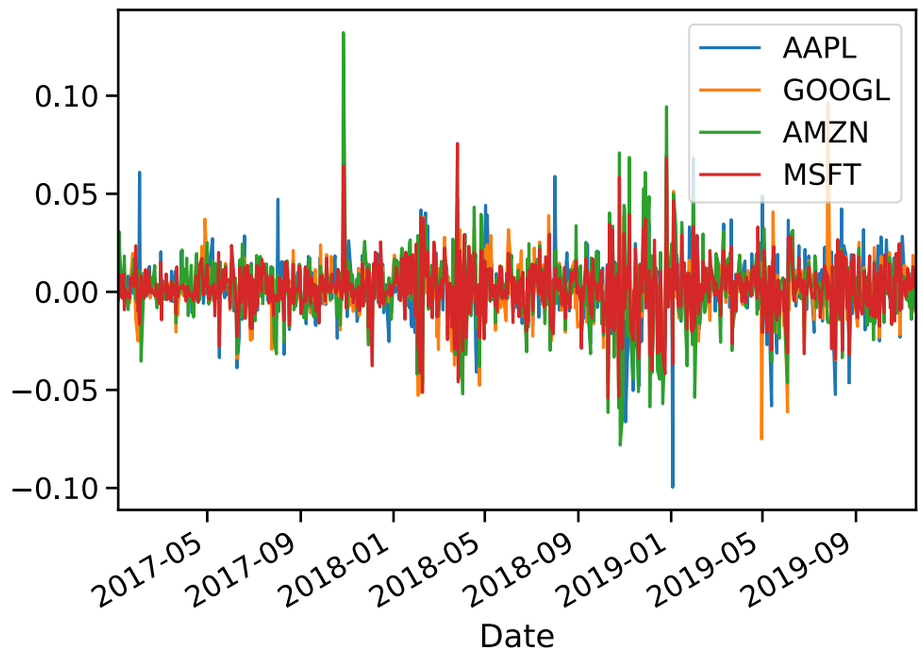
**Prior**

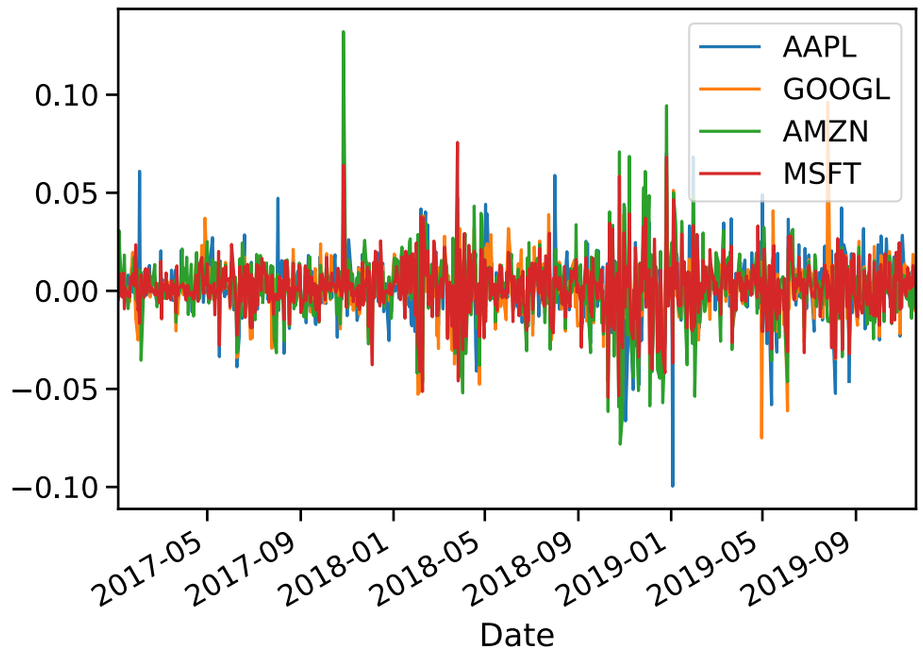



**Posterior**

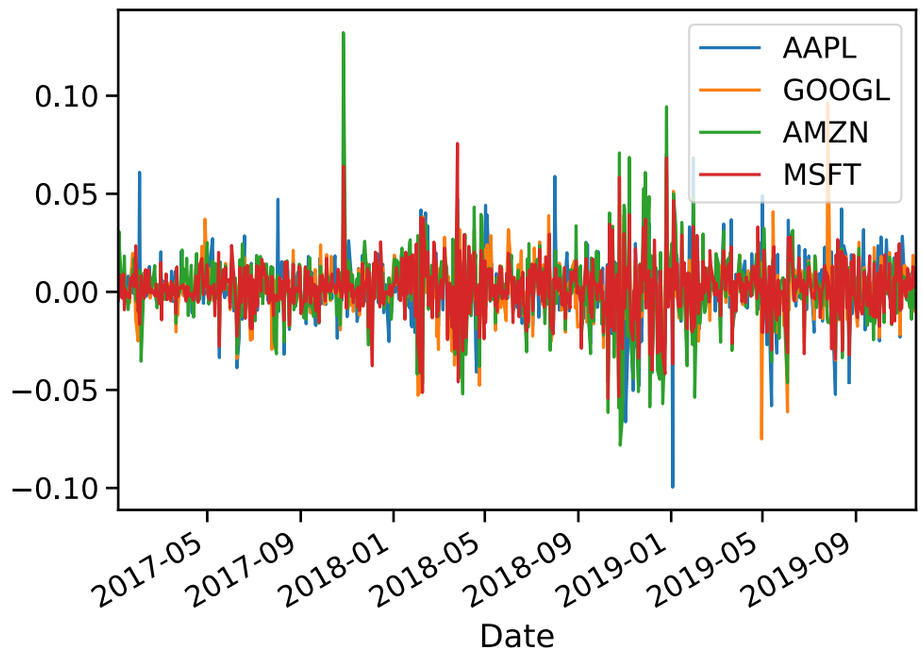



AAPL
GOOGL
AMZN
MSFT
2.75
2.50
2.25
2.00
1.75
1.50
1.25
1.00
2017-05
2017-09
2018-01
2018-05
2018-09
2019-01
2019-05
2019-09
Date

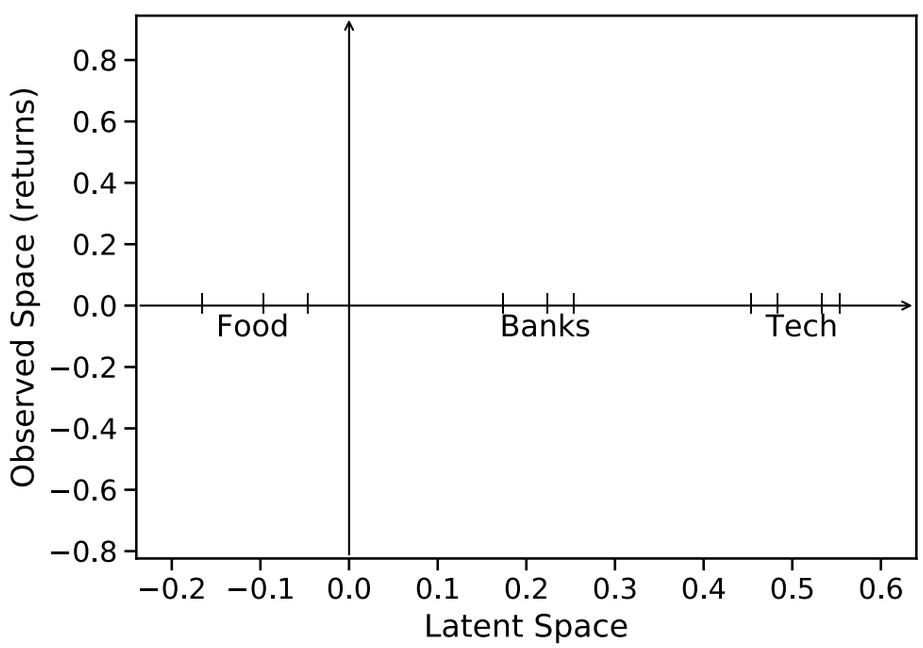

AAPL
GOOGL
AMZN
MSFT
0.10
0.05
0.00
−0.05
−0.10
2017-05
2017-09
2018-01
2018-05
2018-09
2019-01
2019-05
2019-09
Date

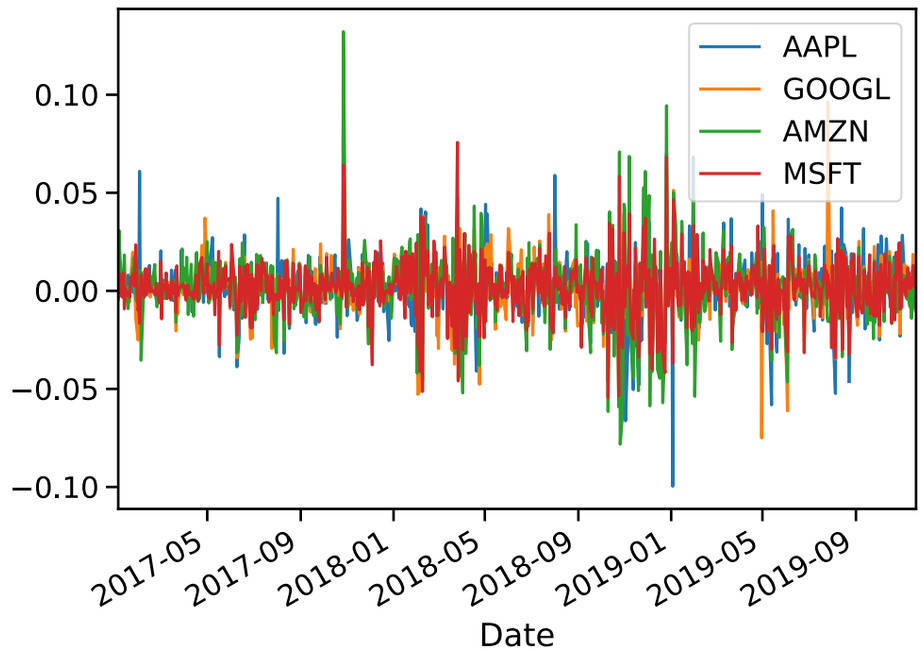



| RETURNS | 24.10.19 | 25.10.19 | 28.10.19 | 29.10.19 |
|---|---|---|---|---|
| AAPL | 0.16 | 1.23 | 1.00 | -2.31 |
| GOOGL | 0.11 | 0.41 | 1.95 | -2.19 |
| AMZN | 1.05 | -1.09 | 0.89 | -0.80 |
| MSFT | 1.96 | 0.56 | 2.45 | -0.94 |
| FOOD | | | | |
| BANK | | | | |

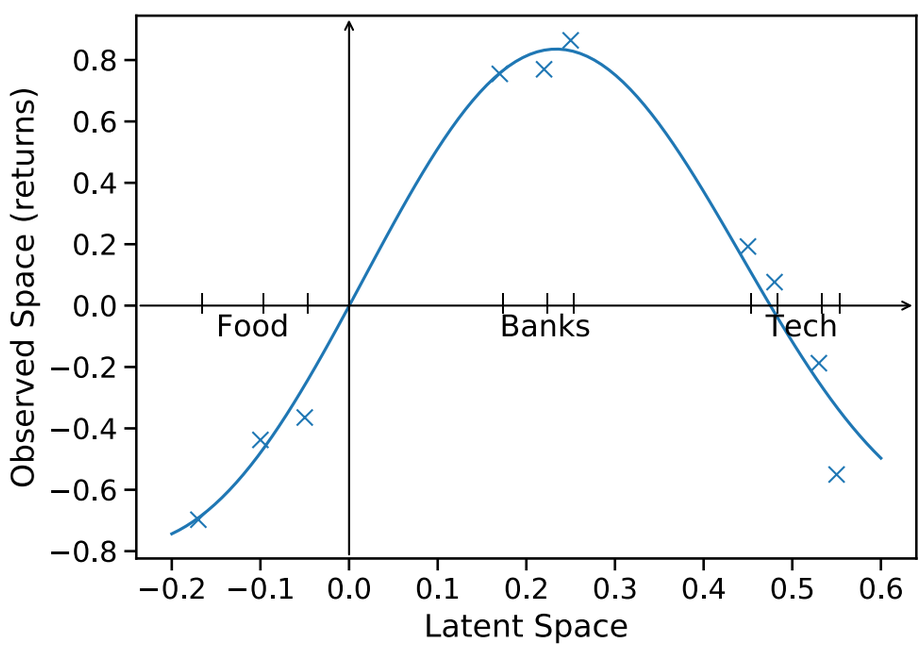


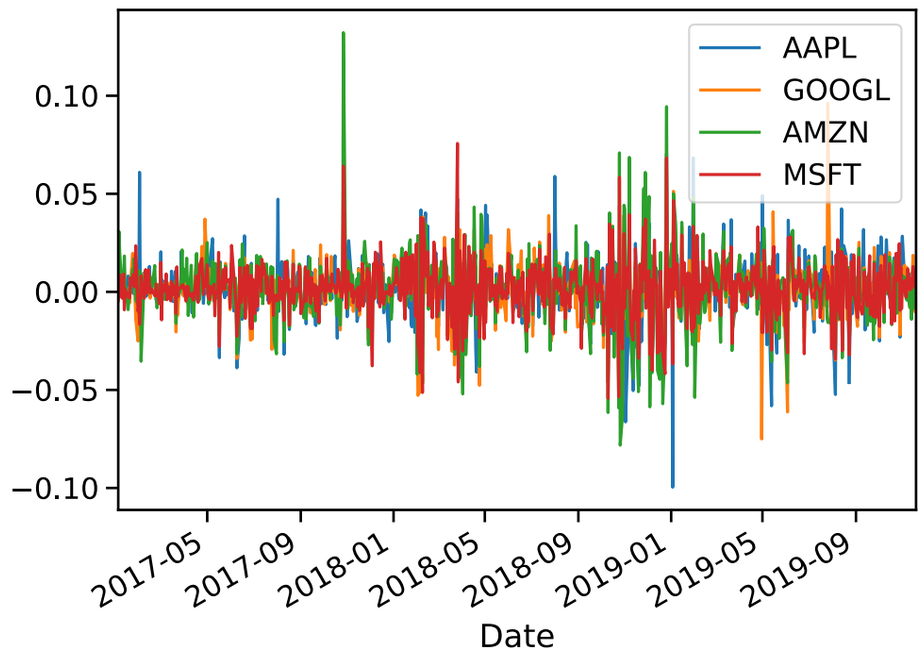

| RETURNS | 24.10.19 | 25.10.19 | 28.10.19 | 29.10.19 |
|---|---|---|---|---|
| AAPL | 0.16 | 1.23 | 1.00 | -2.31 |
| GOOGL | 0.11 | 0.41 | 1.95 | -2.19 |
| AMZN | 1.05 | -1.09 | 0.89 | -0.80 |
| MSFT | 1.96 | 0.56 | 2.45 | -0.94 |
| FOOD | | | | |
| BANK | | | | |



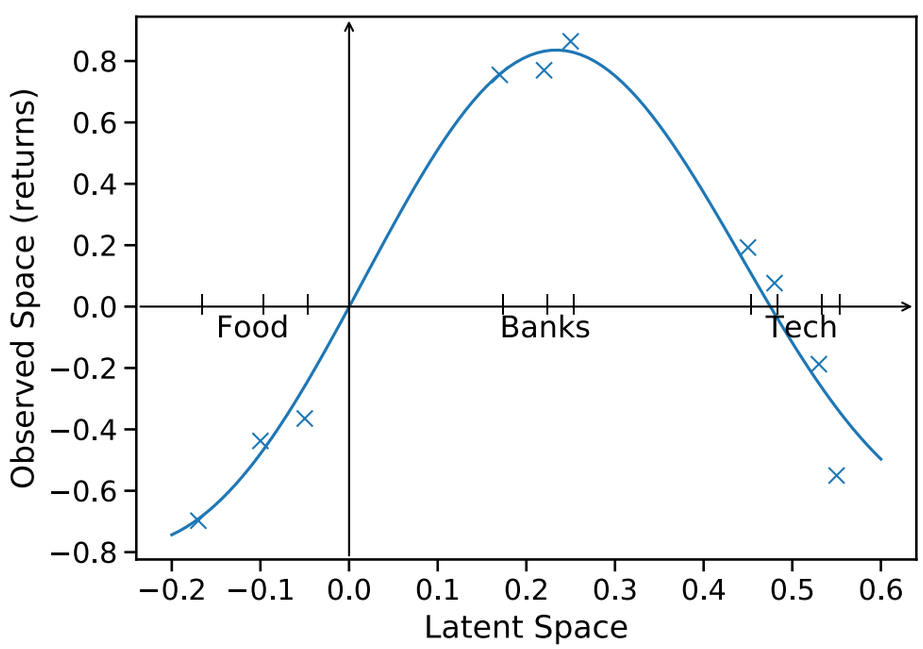



| RETURNS | 24.10.19 | 25.10.19 | 28.10.19 | 29.10.19 |
|---|---|---|---|---|
| AAPL | 0.16 | 1.23 | 1.00 | -2.31 |
| GOOGL | 0.11 | 0.41 | 1.95 | -2.19 |
| AMZN | 1.05 | -1.09 | 0.89 | -0.80 |
| MSFT | 1.96 | 0.56 | 2.45 | -0.94 |
| FOOD | | | | |
| BANK | | | | |



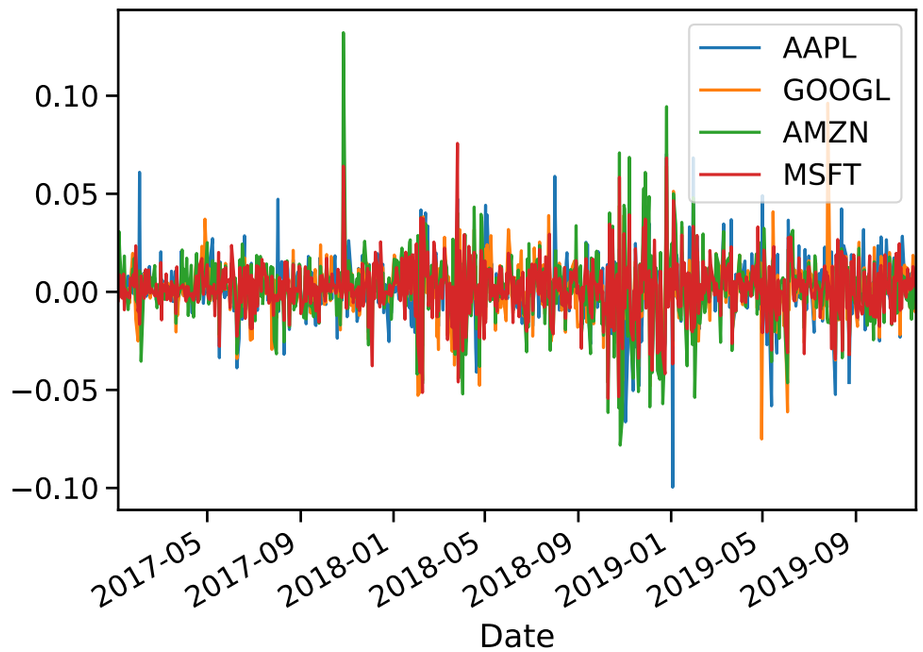



| RETURNS | 24.10.19 | 25.10.19 | 28.10.19 | 29.10.19 |
|---|---|---|---|---|
| AAPL | 0.16 | 1.23 | 1.00 | -2.31 |
| GOOGL | 0.11 | 0.41 | 1.95 | -2.19 |
| AMZN | 1.05 | -1.09 | 0.89 | -0.80 |
| MSFT | 1.96 | 0.56 | 2.45 | -0.94 |
| FOOD | | | | |
| BANK | | | | |

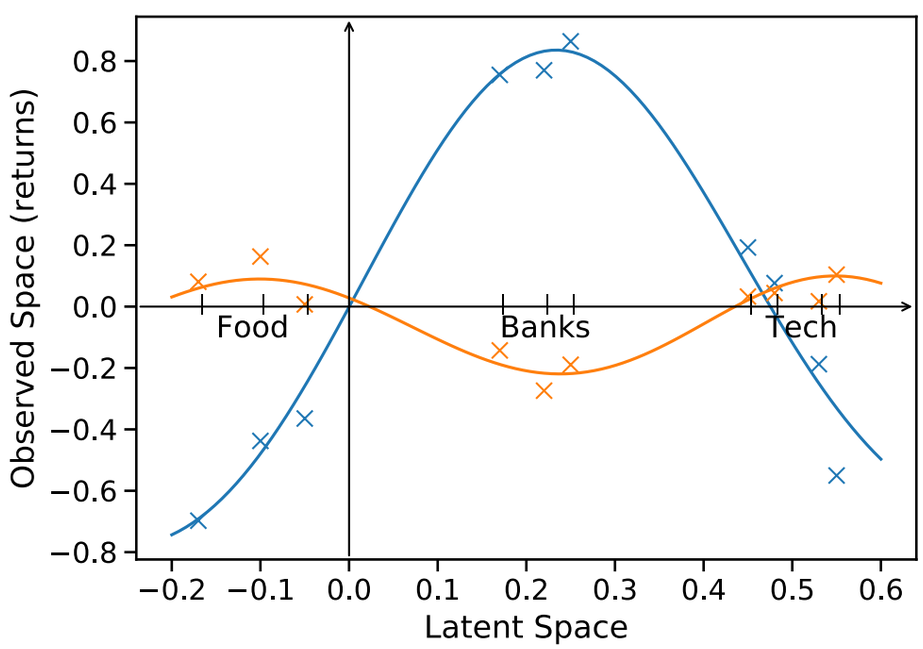



Generative Model

| | Day1 | Day2 |
|---|---|---|
| Bank1 | -0.70 | |
| Bank2 | -0.44 | |
| Bank3 | -0.36 | |
| Food1 | 0.75 | |
| Food2 | 0.77 | |
| Food3 | 0.86 | |
| Tech1 | 0.19 | |
| Tech2 | 0.08 | |
| Tech3 | -0.19 | |
| Tech4 | -0.55 | |

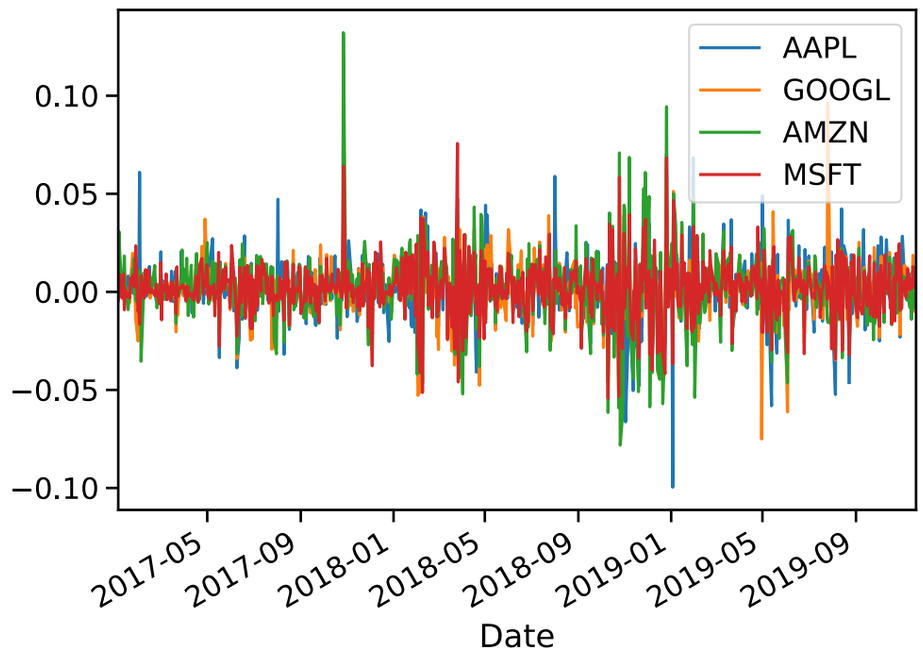



| RETURNS | 24.10.19 | 25.10.19 | 28.10.19 | 29.10.19 |
|---|---|---|---|---|
| AAPL | 0.16 | 1.23 | 1.00 | -2.31 |
| GOOGL | 0.11 | 0.41 | 1.95 | -2.19 |
| AMZN | 1.05 | -1.09 | 0.89 | -0.80 |
| MSFT | 1.96 | 0.56 | 2.45 | -0.94 |
| FOOD | | | | |
| BANK | | | | |

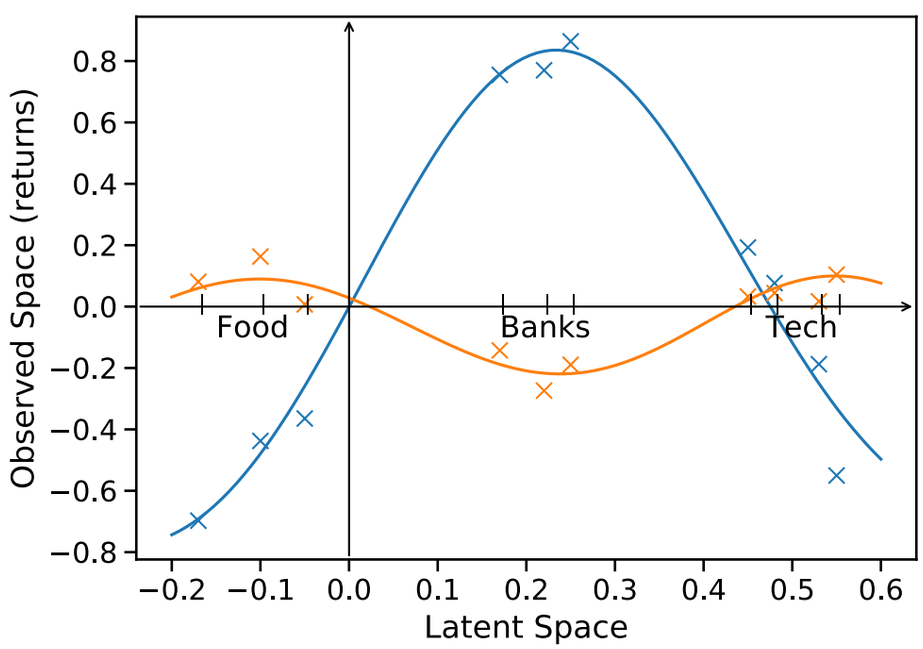



Generative Model

| | Day1 | Day2 |
|---|---|---|
| Bank1 | -0.70 | |
| Bank2 | -0.44 | |
| Bank3 | -0.36 | |
| Food1 | 0.75 | |
| Food2 | 0.77 | |
| Food3 | 0.86 | |
| Tech1 | 0.19 | |
| Tech2 | 0.08 | |
| Tech3 | -0.19 | |
| Tech4 | -0.55 | |

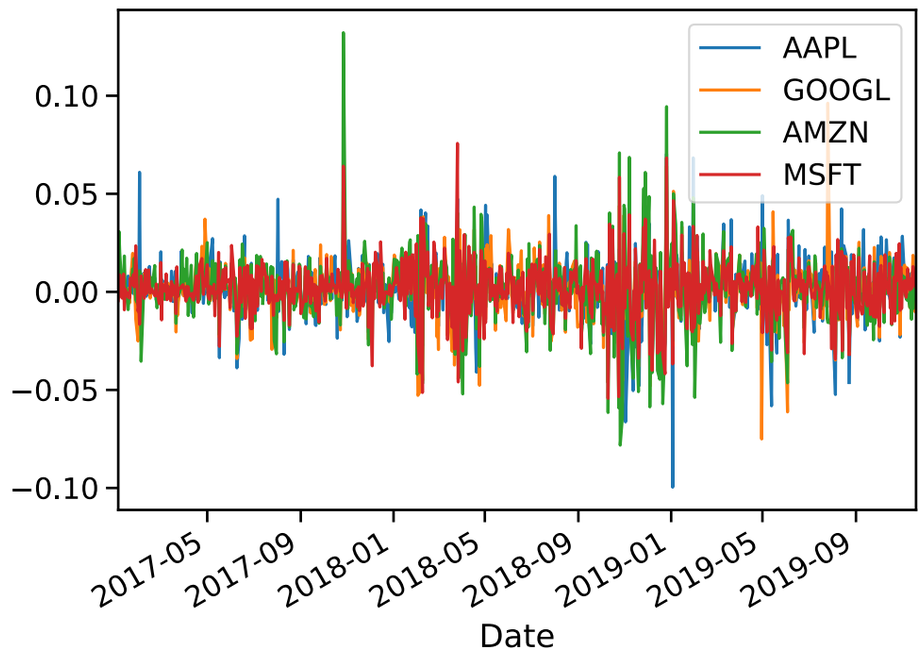



| RETURNS | 24.10.19 | 25.10.19 | 28.10.19 | 29.10.19 |
|---|---|---|---|---|
| AAPL | 0.16 | 1.23 | 1.00 | -2.31 |
| GOOGL | 0.11 | 0.41 | 1.95 | -2.19 |
| AMZN | 1.05 | -1.09 | 0.89 | -0.80 |
| MSFT | 1.96 | 0.56 | 2.45 | -0.94 |
| FOOD | | | | |
| BANK | | | | |

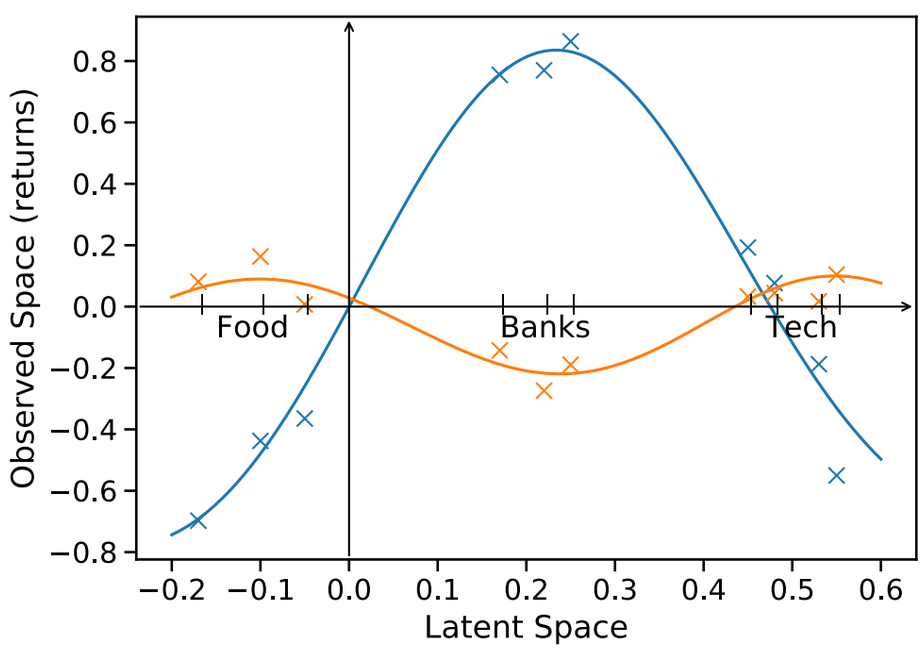



Generative Model

| | Day1 | Day2 |
|---|---|---|
| Bank1 | -0.70 | 0.08 |
| Bank2 | -0.44 | 0.16 |
| Bank3 | -0.36 | 0.01 |
| Food1 | 0.75 | -0.14 |
| Food2 | 0.77 | -0.27 |
| Food3 | 0.86 | -0.19 |
| Tech1 | 0.19 | 0.03 |
| Tech2 | 0.08 | 0.04 |
| Tech3 | -0.19 | 0.02 |
| Tech4 | -0.55 | 0.10 |




| RETURNS | 24.10.19 | 25.10.19 | 28.10.19 | 29.10.19 |
|---|---|---|---|---|
| AAPL | 0.16 | 1.23 | 1.00 | -2.31 |
| GOOGL | 0.11 | 0.41 | 1.95 | -2.19 |
| AMZN | 1.05 | -1.09 | 0.89 | -0.80 |
| MSFT | 1.96 | 0.56 | 2.45 | -0.94 |
| FOOD | | | | |
| BANK | | | | |




Generative Model

Learning/Inference

| | Day1 | Day2 |
|---|---|---|
| Bank1 | -0.70 | 0.08 |
| Bank2 | -0.44 | 0.16 |
| Bank3 | -0.36 | 0.01 |
| Food1 | 0.75 | -0.14 |
| Food2 | 0.77 | -0.27 |
| Food3 | 0.86 | -0.19 |
| Tech1 | 0.19 | 0.03 |
| Tech2 | 0.08 | 0.04 |
| Tech3 | -0.19 | 0.02 |
| Tech4 | -0.55 | 0.10 |

# GPLVM as extension to APT

Capital Assert Pricing Model (CAPM)

$$r_n = r_m \beta_n + \epsilon_n$$

$$r_{n,d} = r_{m,d} \beta_n + \epsilon_{n,d}$$

# GPLVM as extension to APT

Capital Assert Pricing Model (CAPM)

$$r_n = r_m \beta_n + \epsilon_n$$

$$r_{n,d} = r_{m,d} \beta_n + \epsilon_{n,d}$$

Arbitrage theory of capital asset pricing (APT)

$$\boldsymbol{r} = \boldsymbol{B}\boldsymbol{F}^T + \boldsymbol{\epsilon}$$

$$\boldsymbol{r} \in \mathbb{R}^{N \times D}$$
$$\boldsymbol{B} \in \mathbb{R}^{N \times Q}$$
$$\boldsymbol{F} \in \mathbb{R}^{D \times Q}$$

$$\boldsymbol{F} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}), \quad \boldsymbol{\epsilon} \sim \mathcal{N}\left(\mathbf{0}, \boldsymbol{\Psi}\right)$$

$$p(\boldsymbol{r} \,|\, \boldsymbol{B}) = \prod_{d=1}^{D} \mathcal{N}(\boldsymbol{r}_{:,d} \,|\, \mathbf{0}, \boldsymbol{B}\boldsymbol{B}^T + \boldsymbol{\Psi})$$

# GPLVM as extension to APT

Capital Assert Pricing Model (CAPM)

$$r_n = r_m \beta_n + \epsilon_n$$

$$r_{n,d} = r_{m,d} \beta_n + \epsilon_{n,d}$$

Arbitrage theory of capital asset pricing (APT)

$$\boldsymbol{r} = \boldsymbol{B}\boldsymbol{F}^T + \boldsymbol{\epsilon}$$

$$\boldsymbol{r} \in \mathbb{R}^{N \times D}$$
$$\boldsymbol{B} \in \mathbb{R}^{N \times Q}$$
$$\boldsymbol{F} \in \mathbb{R}^{D \times Q}$$

$$\boldsymbol{F} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}), \quad \boldsymbol{\epsilon} \sim \mathcal{N}\left(\mathbf{0}, \boldsymbol{\Psi}\right)$$

$$p(\boldsymbol{r} \,|\, \boldsymbol{B}) = \prod_{d=1}^{D} \mathcal{N}(\boldsymbol{r}_{:,d} \,|\, \mathbf{0}, \boldsymbol{B}\boldsymbol{B}^T + \boldsymbol{\Psi})$$

Gaussian process latent variable model (GPLVM)

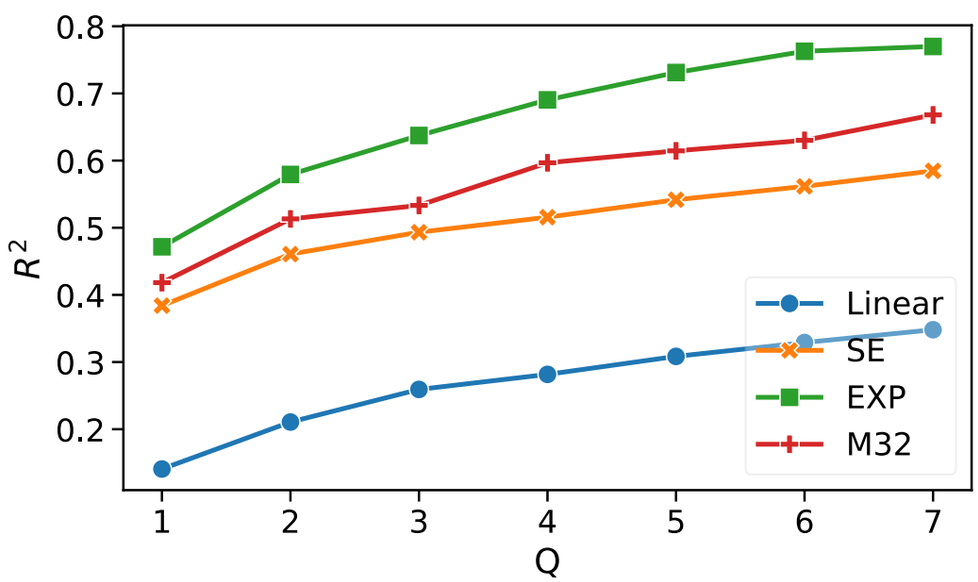
$$p(\boldsymbol{r} \,|\, \boldsymbol{B}) = \prod_{d=1}^{D} \mathcal{N}(\boldsymbol{r}_{:,d} \,|\, \mathbf{0}, \boldsymbol{K}(\boldsymbol{B}, \boldsymbol{B}) + \boldsymbol{\Psi})$$

$$\boldsymbol{r}_{n,d} = f_d(\boldsymbol{B}_{n,:}) + \boldsymbol{\epsilon}_{n,d}$$

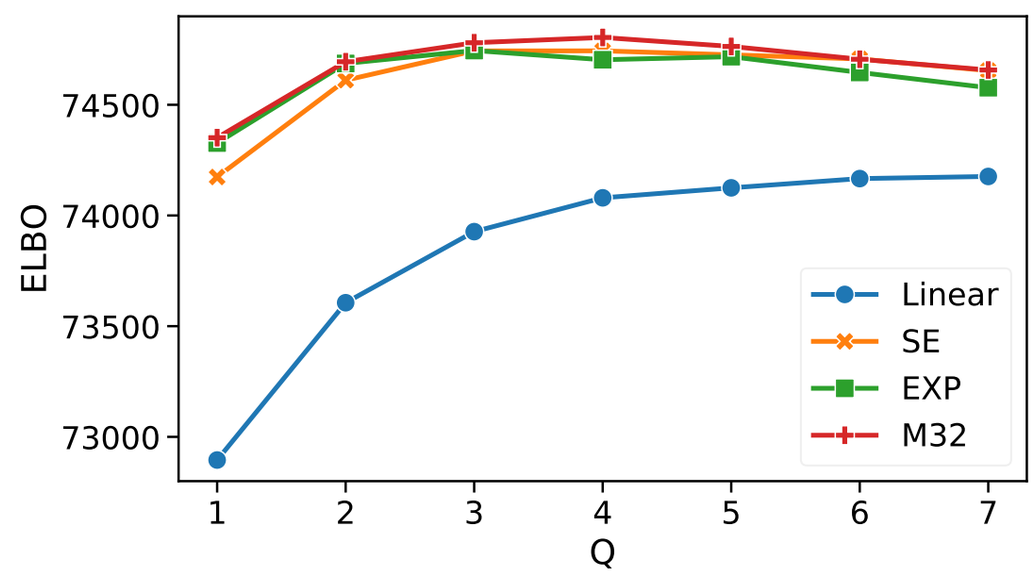
# Experiments

**Use Variational Bayes for the inference - data $Y \in \mathbb{R}^{N \times D}$**
Approximate the true posterior with a simple distribution

$R^2$ - Variance of the data captured by the model



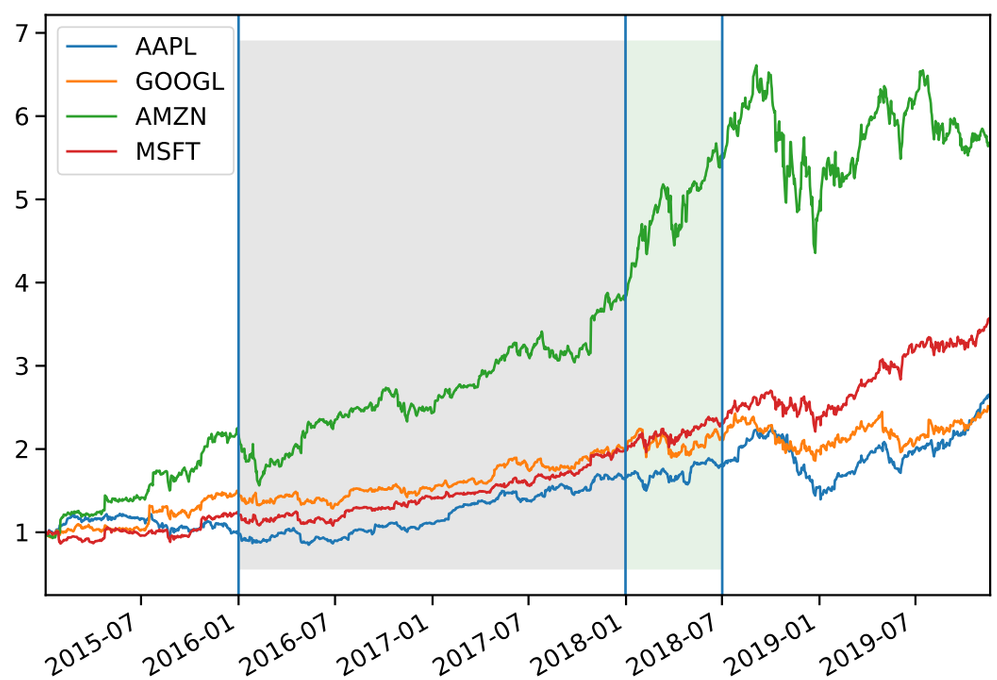

**ELBO** - Lower Bound to the marginal likelihood



# Experiments

**Portfolio Allocation** - Backtested on S&P500 from 2002 to 2018

| Model | Linear | SE | EXP | M32 | Sample Cov | Ledoit Wolf | Eq. Weighted |
|---|---|---|---|---|---|---|---|
| Mean | 0.142 | 0.151 | 0.155 | 0.158 | 0.149 | 0.148 | 0.182 |
| Std | 0.158 | 0.156 | 0.154 | 0.153 | 0.159 | 0.159 | 0.232 |
| Sharpe ratio | 0.901 | 0.969 | 1.008 | 1.029 | 0.934 | 0.931 | 0.786 |



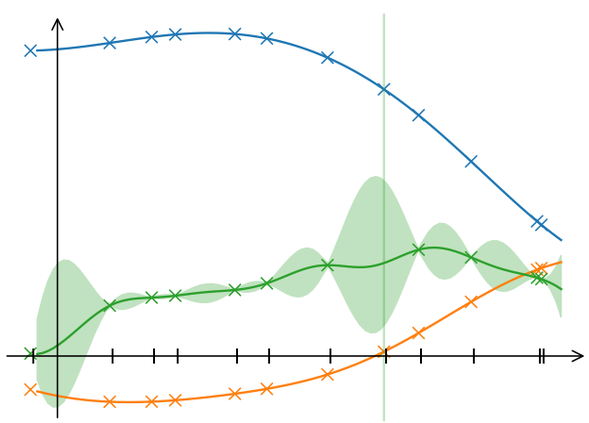
# Experiments

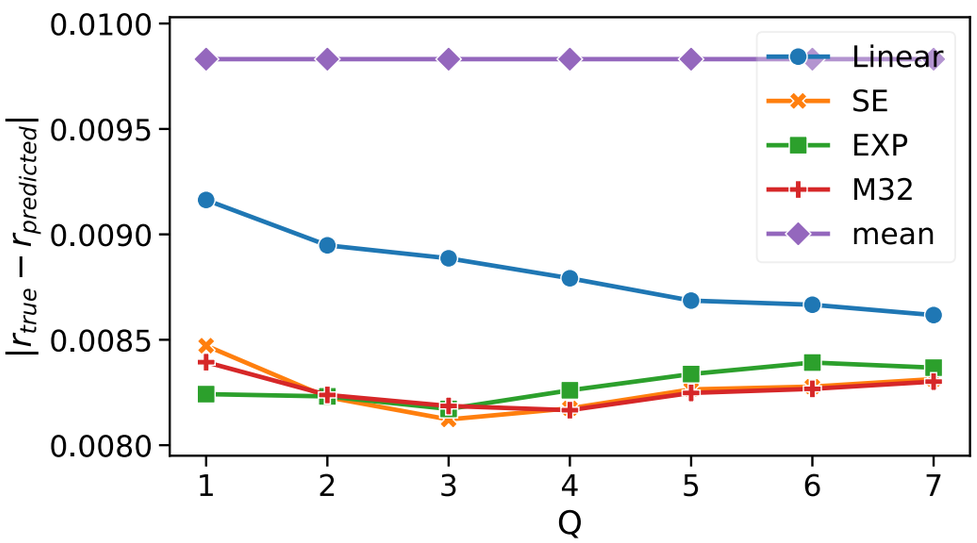
**Portfolio Allocation** - Backtested on S&P500 from 2002 to 2018

| Model | Linear | SE | EXP | M32 | Sample Cov | Ledoit Wolf | Eq. Weighted |
|---|---|---|---|---|---|---|---|
| Mean | 0.142 | 0.151 | 0.155 | 0.158 | 0.149 | 0.148 | 0.182 |
| Std | 0.158 | 0.156 | 0.154 | 0.153 | 0.159 | 0.159 | 0.232 |
| Sharpe ratio | 0.901 | 0.969 | 1.008 | 1.029 | 0.934 | 0.931 | 0.786 |

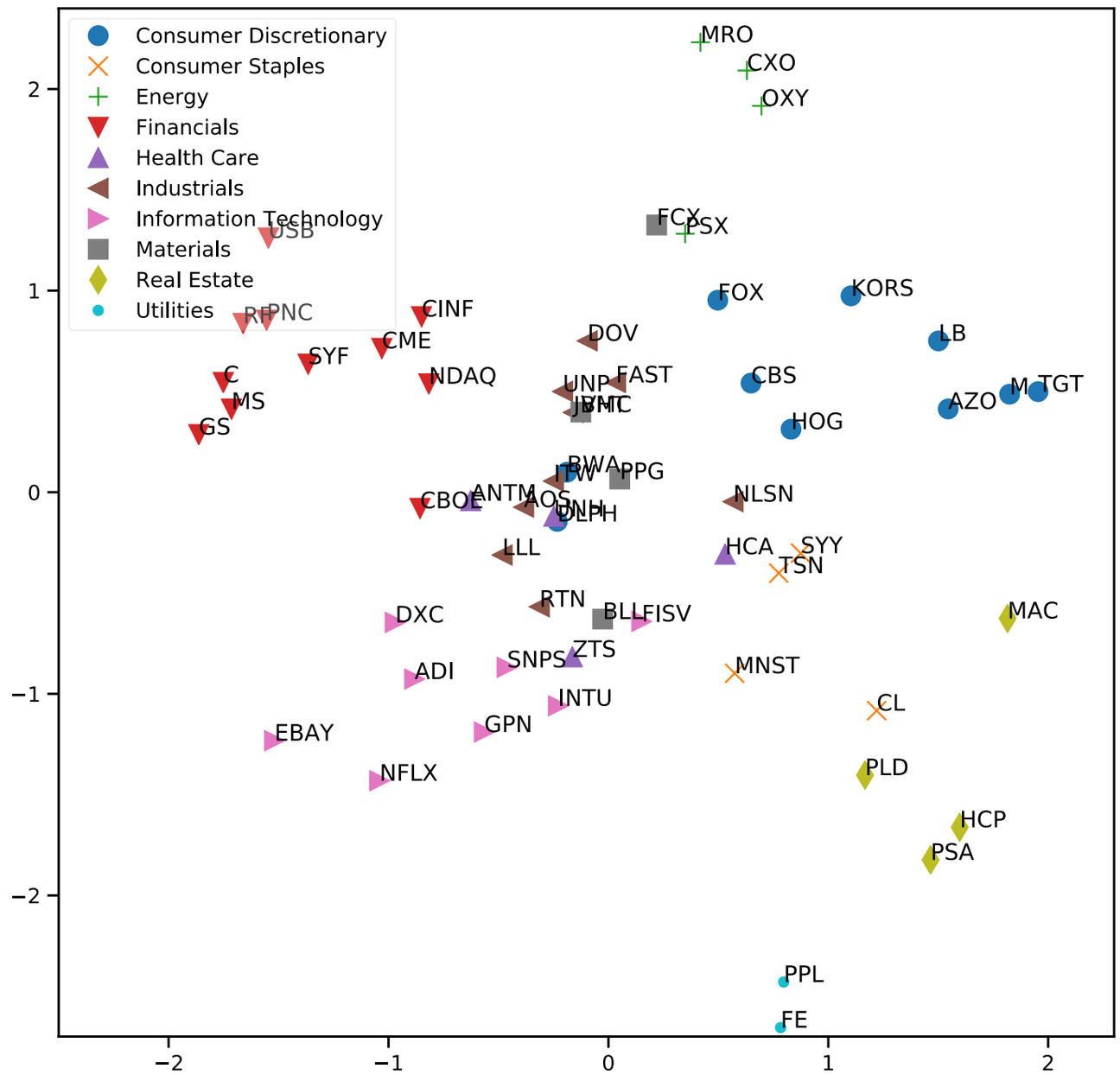
**Prediction of missing values** - on held out dataset (now casting, data imputation)

$$\begin{pmatrix} & a_1 & a_2 & a_3 \\ & b_1 & b_2 & NaN \\ \cdots & c_1 & c_2 & c_3 \\ & d_1 & d_2 & d_3 \end{pmatrix}$$





# Visualization of Latent Space



# Visualization of Latent Space

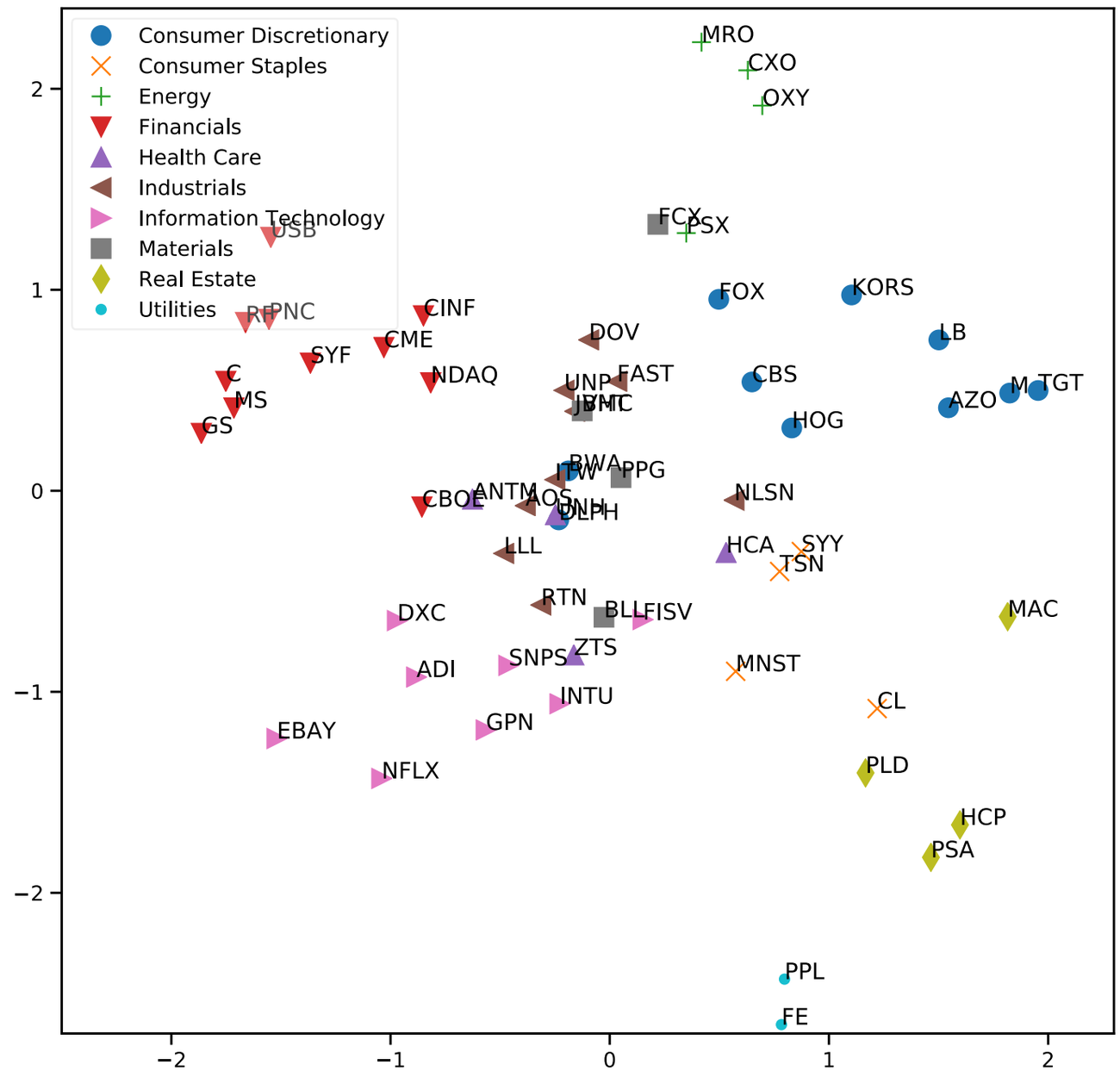
## Rotational Symmetry

$$k_{linear}(x, x') = xx'$$

$$k_{rbf}(x, x') = \exp\left(-\frac{1}{2\ell^2}(x - x')^2\right)$$

$$k_{ou}(x, x') = \exp\left(-\frac{1}{\ell}|x - x'|\right)$$

$$k_{periodic}(x, x') = \exp\left(-\frac{2}{\ell^2}\sin^2(|x - x'|)\right)$$



# Probabilistic PCA

## Probabilistic PCA (PPCA)

Viewed as a generative model, that maps the latent space $\boldsymbol{X}$ to the data space $\boldsymbol{Y}$

$$\boldsymbol{X} \in \mathbb{R}^{N \times Q} \quad \rightarrow \quad \boldsymbol{Y} \in \mathbb{R}^{N \times D}$$

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{W}^T + \boldsymbol{\epsilon}$$

$$\boldsymbol{X} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}), \quad \epsilon \sim \mathcal{N}\left(\boldsymbol{0}, \sigma^2 \boldsymbol{I}\right)$$

$$p(\boldsymbol{Y} \mid \boldsymbol{W}) = \prod_{n=1}^{N} \mathcal{N}\left(\boldsymbol{Y}_{n,:} \mid \boldsymbol{0}, \boldsymbol{W}\boldsymbol{W}^T + \sigma^2 \boldsymbol{I}\right)$$

$$\boldsymbol{W}\boldsymbol{R}\boldsymbol{R}^T\boldsymbol{W}^T = \boldsymbol{W}\boldsymbol{W}^T \quad \forall\ \boldsymbol{R}\boldsymbol{R}^T = \boldsymbol{I}$$

# Probabilistic PCA

## Probabilistic PCA (PPCA)

Viewed as a generative model, that maps the latent space $\boldsymbol{X}$ to the data space $\boldsymbol{Y}$

$$\boldsymbol{X} \in \mathbb{R}^{N \times Q} \quad \rightarrow \quad \boldsymbol{Y} \in \mathbb{R}^{N \times D}$$

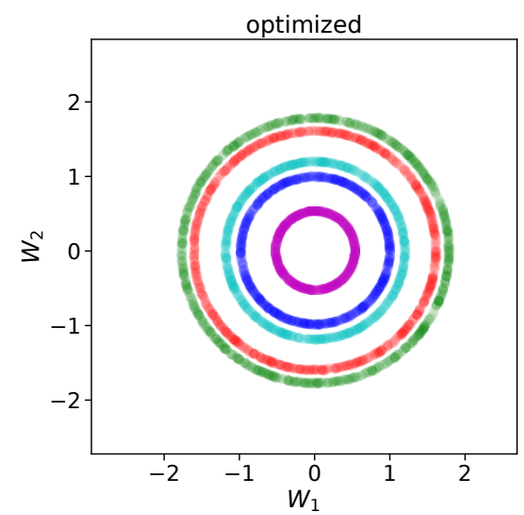
$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{W}^T + \boldsymbol{\epsilon}$$

$$\boldsymbol{X} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}), \quad \epsilon \sim \mathcal{N}\left(\boldsymbol{0}, \sigma^2 \boldsymbol{I}\right)$$

$$p(\boldsymbol{Y} \mid \boldsymbol{W}) = \prod_{n=1}^{N} \mathcal{N}\left(\boldsymbol{Y}_{n,:} \mid \boldsymbol{0}, \boldsymbol{W}\boldsymbol{W}^T + \sigma^2 \boldsymbol{I}\right)$$

$$\boldsymbol{W}\boldsymbol{R}\boldsymbol{R}^T\boldsymbol{W}^T = \boldsymbol{W}\boldsymbol{W}^T \quad \forall\ \boldsymbol{R}\boldsymbol{R}^T = \boldsymbol{I}$$

- Rotation invariant model




$$\begin{pmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \\ w_{31} & w_{32} \\ w_{41} & w_{42} \\ w_{51} & w_{52} \end{pmatrix} \in \mathbb{R}^{5 \times 2}$$

# Probabilistic PCA

## Probabilistic PCA (PPCA)

Viewed as a generative model, that maps the latent space $\boldsymbol{X}$ to the data space $\boldsymbol{Y}$

$$\boldsymbol{X} \in \mathbb{R}^{N \times Q} \quad \rightarrow \quad \boldsymbol{Y} \in \mathbb{R}^{N \times D}$$

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{W}^T + \boldsymbol{\epsilon}$$

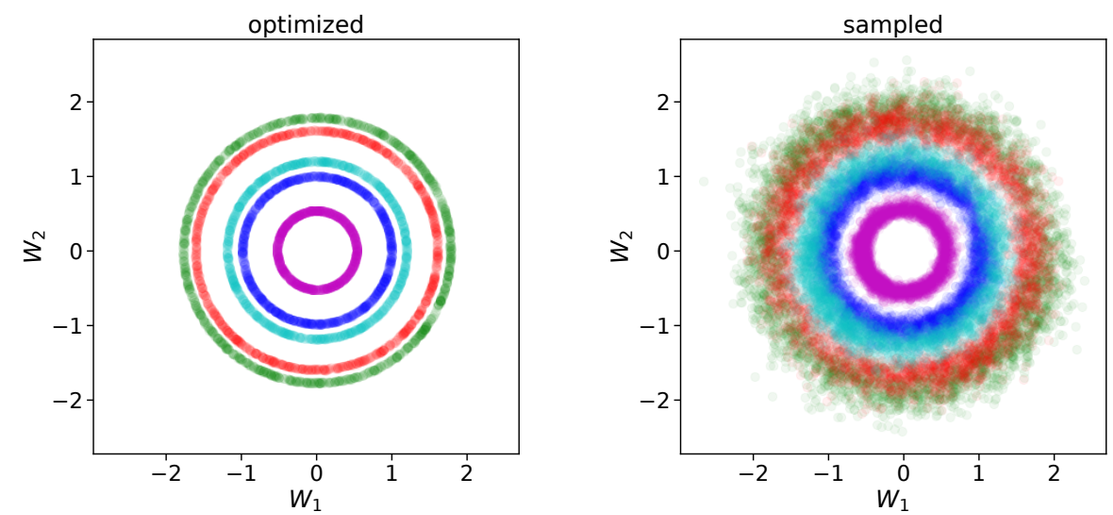
$$\boldsymbol{X} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}), \quad \epsilon \sim \mathcal{N}\left(\boldsymbol{0}, \sigma^2 \boldsymbol{I}\right)$$

$$p(\boldsymbol{Y} \mid \boldsymbol{W}) = \prod_{n=1}^{N} \mathcal{N}\left(\boldsymbol{Y}_{n,:} \mid \boldsymbol{0}, \boldsymbol{W}\boldsymbol{W}^T + \sigma^2 \boldsymbol{I}\right)$$

$$\boldsymbol{W}\boldsymbol{R}\boldsymbol{R}^T\boldsymbol{W}^T = \boldsymbol{W}\boldsymbol{W}^T \quad \forall\, \boldsymbol{R}\boldsymbol{R}^T = \boldsymbol{I}$$

$$p\left(\boldsymbol{W} \mid \boldsymbol{Y}\right) = \frac{p\left(\boldsymbol{Y} \mid \boldsymbol{W}\right) p\left(\boldsymbol{W}\right)}{p(\boldsymbol{Y})}$$

- Rotation invariant model




$$\begin{pmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \\ w_{31} & w_{32} \\ w_{41} & w_{42} \\ w_{51} & w_{52} \end{pmatrix} \in \mathbb{R}^{5 \times 2}$$

# Solution

**Outline of procedure**

- SVD of W $\boldsymbol{W}\boldsymbol{W}^T = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T\left(\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T\right)^T = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T\boldsymbol{V}\boldsymbol{\Sigma}^T\boldsymbol{U}^T = \boldsymbol{U}\boldsymbol{\Sigma}^2\boldsymbol{U}^T$
- Fix coordinate system $\boldsymbol{V} = \boldsymbol{I}$
- Specify prior $p(\boldsymbol{U}, \boldsymbol{\Sigma})$
- Sample from $p(\boldsymbol{U}, \boldsymbol{\Sigma} \mid \boldsymbol{Y})$

# Solution

**Outline of procedure**

- SVD of W $\quad \boldsymbol{W}\boldsymbol{W}^T = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T\left(\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T\right)^T = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T\boldsymbol{V}\boldsymbol{\Sigma}^T\boldsymbol{U}^T = \boldsymbol{U}\boldsymbol{\Sigma}^2\boldsymbol{U}^T$
- Fix coordinate system $\quad \boldsymbol{V} = \boldsymbol{I}$
- Specify prior $\quad p(\boldsymbol{U}, \boldsymbol{\Sigma})$
- Sample from $\quad p(\boldsymbol{U}, \boldsymbol{\Sigma} \mid \boldsymbol{Y})$

$$p(\boldsymbol{Y} \mid \boldsymbol{W}) = \prod_{n=1}^{N} \mathcal{N}\left(\boldsymbol{Y}_{n,:} \mid \boldsymbol{0}, \boldsymbol{W}\boldsymbol{W}^T + \sigma^2\boldsymbol{I}\right)$$

$$p(\boldsymbol{Y} \mid \boldsymbol{U}, \boldsymbol{\Sigma}) = \prod_{n=1}^{N} \mathcal{N}\left(\boldsymbol{Y}_{n,:} \mid \boldsymbol{0}, \boldsymbol{U}\boldsymbol{\Sigma}^2\boldsymbol{U}^T + \sigma^2\boldsymbol{I}\right)$$

$\boldsymbol{W} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}) \quad \rightarrow \quad \boldsymbol{W}\boldsymbol{W}^T$ **is Wishart distributed**

$\boldsymbol{U} \sim ?$
$\boldsymbol{\Sigma} \sim ?$ $\quad \rightarrow \quad \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{\Sigma}^T\boldsymbol{U}^T$ **is Wishart distributed**

# Solution

**Outline of procedure**

- SVD of W $\quad \boldsymbol{W}\boldsymbol{W}^T = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T\left(\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T\right)^T = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T\boldsymbol{V}\boldsymbol{\Sigma}^T\boldsymbol{U}^T = \boldsymbol{U}\boldsymbol{\Sigma}^2\boldsymbol{U}^T$
- Fix coordinate system $\quad \boldsymbol{V} = \boldsymbol{I}$
- Specify prior $\quad p(\boldsymbol{U}, \boldsymbol{\Sigma})$
- Sample from $\quad p(\boldsymbol{U}, \boldsymbol{\Sigma} \,|\, \boldsymbol{Y})$

$$p(\boldsymbol{Y} \,|\, \boldsymbol{W}) = \prod_{n=1}^{N} \mathcal{N}\left(\boldsymbol{Y}_{n,:} \,|\, \boldsymbol{0}, \boldsymbol{W}\boldsymbol{W}^T + \sigma^2\boldsymbol{I}\right)$$

$$p(\boldsymbol{Y} \,|\, \boldsymbol{U}, \boldsymbol{\Sigma}) = \prod_{n=1}^{N} \mathcal{N}\left(\boldsymbol{Y}_{n,:} \,|\, \boldsymbol{0}, \boldsymbol{U}\boldsymbol{\Sigma}^2\boldsymbol{U}^T + \sigma^2\boldsymbol{I}\right)$$

$\boldsymbol{W} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}) \quad \rightarrow \quad \boldsymbol{W}\boldsymbol{W}^T$ **is Wishart distributed**

$\boldsymbol{U} \sim ?$, $\boldsymbol{\Sigma} \sim ?$ $\quad \rightarrow \quad \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{\Sigma}^T\boldsymbol{U}^T$ **is Wishart distributed**

---

$\rightarrow$ $\boldsymbol{U}$ is uniformly distributed in the space of orthogonal matrices (Stiefel manifold)

$$\rightarrow \qquad p\left(\sigma_1, \ldots, \sigma_Q\right) = c e^{-\frac{1}{2}\sum_{q=1}^{Q}\sigma_q^2} \prod_{q=1}^{Q}\left(\sigma_q^{D-Q-1} \prod_{q'=q+1}^{Q}\left|\sigma_q^2 - \sigma_{q'}^2\right|\right)\prod_{q=1}^{Q} 2\sigma_q$$

James & Lee (2014)

# Implementation

## How to uniformly sample $U$ on $\mathscr{V}_{Q,D}$

**Theorem 2** *Let $\boldsymbol{v}_D, \boldsymbol{v}_{D-1}, \ldots, \boldsymbol{v}_1$ be uniformly distributed on the unit spheres $\mathbb{S}^{D-1}, \ldots, \mathbb{S}^0$ respectively, where $\mathbb{S}^{n-1}$ is the unit sphere in $\mathbb{R}^n$. Furthermore, let $\boldsymbol{H}_n(\boldsymbol{v}_n)$ be the $n$-th Householder transformation as defined in equation (2.20) The product*

$$\boldsymbol{Q} = \boldsymbol{H}_D(\boldsymbol{v}_D)\boldsymbol{H}_{D-1}(\boldsymbol{v}_{D-1})\ldots\boldsymbol{H}_1(\boldsymbol{v}_1) \qquad (2.21)$$

*is a random orthogonal matrix with distribution given by the Haar measure on $O(D)$.*

**Mezzadri (2007)**

**for** $n = D : 1$

$\boldsymbol{v}_n \sim$ **uniform on** $\mathbb{S}^{n-1}$

$$\boldsymbol{u}_n = \frac{\boldsymbol{v}_n + \operatorname{sgn}(\boldsymbol{v}_{n1}) \parallel \boldsymbol{v}_n \parallel \boldsymbol{e}_1}{\left\| \boldsymbol{v}_n + \operatorname{sgn}(\boldsymbol{v}_{n1}) \parallel \boldsymbol{v}_n \parallel \boldsymbol{e}_1 \right\|}$$

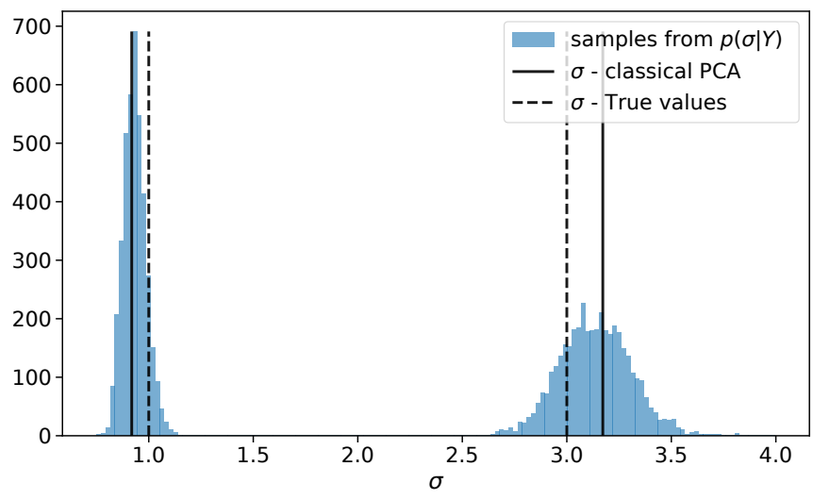
$$\tilde{\boldsymbol{H}}_{\boldsymbol{n}}\left(\boldsymbol{v}_n\right) = -\operatorname{sgn}\left(\boldsymbol{v}_{n1}\right)\left(\boldsymbol{I} - 2\boldsymbol{u}_n\boldsymbol{u}_n^T\right)$$

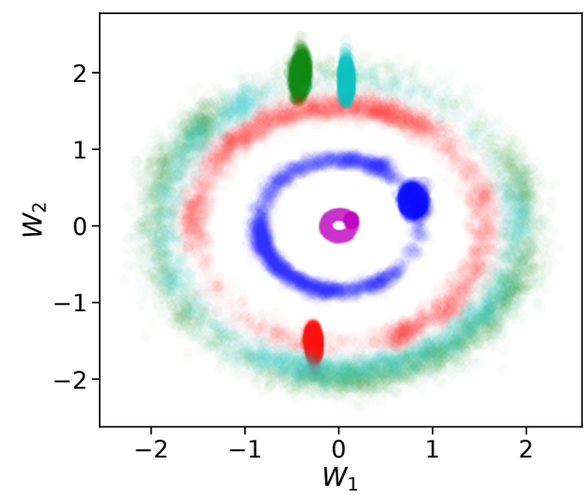
$$\boldsymbol{H}_n = \begin{pmatrix} \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{0} & \tilde{\boldsymbol{H}}_{\boldsymbol{n}} \end{pmatrix}$$

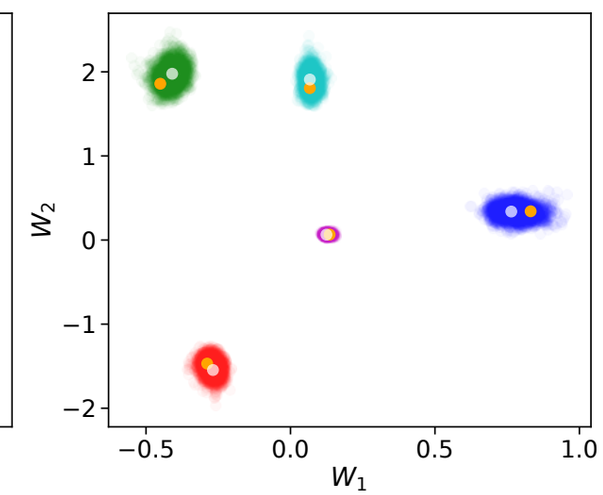
$$\boldsymbol{U} = \boldsymbol{H}_D\left(\boldsymbol{v}_D\right)\boldsymbol{H}_{D-1}\left(\boldsymbol{v}_{D-1}\right)\ldots\boldsymbol{H}_1\left(\boldsymbol{v}_1\right)$$

# Results

## Inference on Synthetic Dataset $(N, D) = (150, 5)$



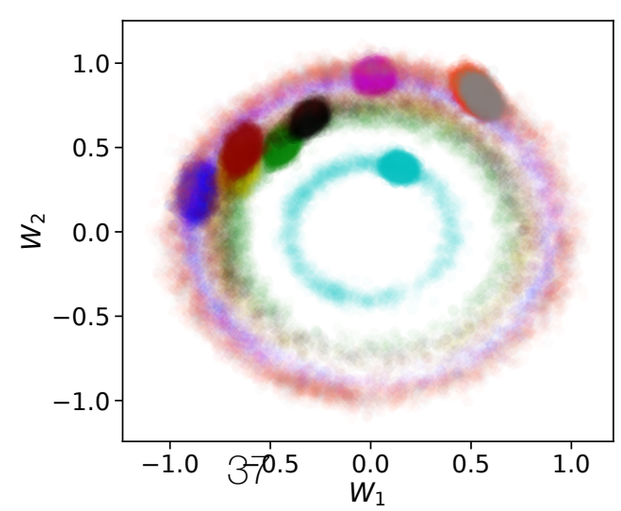

## Breast Cancer Wisconsin Dataset $(N, D) = (569, 30)$

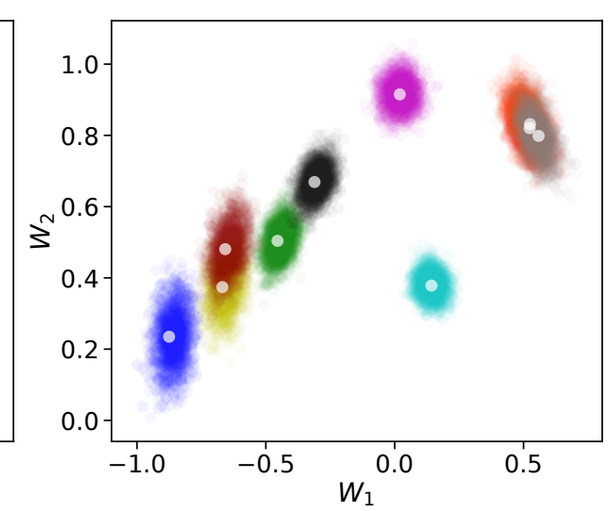
Breast Cancer

$(N, D) = (569,30)$

Time House: 9.5 min

Time Standard: 25.6 min



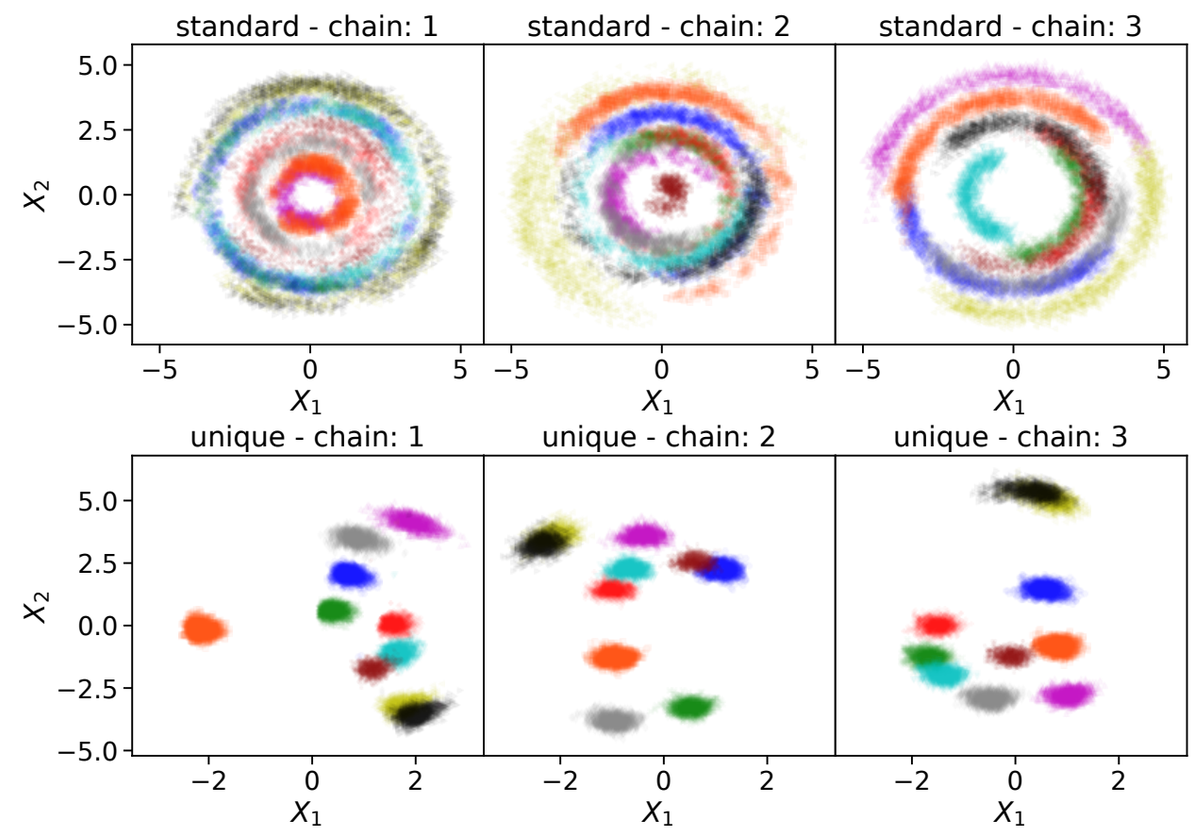



# Extension to non-linear models

- GPLVM with the same rotation invariant problem

$$p(\boldsymbol{Y}|\boldsymbol{X}) = \prod_{d=1}^{D} \mathcal{N}\left(\boldsymbol{Y}_{:,d}|\boldsymbol{\mu}, \boldsymbol{K} + \sigma^2 I\right)$$

$$k_{\mathrm{SE}}(\boldsymbol{x}, \boldsymbol{x}') = \sigma_{\mathrm{SE}}^2 \exp\left(-0.5 \ \| \boldsymbol{x} - \boldsymbol{x}' \|_2^2 / l^2\right)$$



# Benefits of proposed parameterization

- Broke rotational symmetry in latent space without changing the model (standard Gaussian prior on $\boldsymbol{W}$)
- Efficient implementation via Householder transformations (no Jacobian correction needed)
- New parameterization allows for other interpretable priors on rotation and principle variances
- Easy extension to non-linear models

# Bayesian Quantile Matching Estimation

| Country | Sample Size | 25 | 50 | 75 |
|---|---|---|---|---|
| EL | 12918 | 4930 | 7500 | 11000 |
| ES | 19177 | 8803 | 13681 | 20413 |
| FR | 21325 | 16185 | 21713 | 29008 |
| IT | 24969 | 10699 | 16247 | 22944 |
| LU | 10292 | 23964 | 33818 | 48692 |
| NL | 12748 | 16879 | 22733 | 30327 |
| SE | 11635 | 17794 | 25164 | 33365 |
| UK | 17645 | 14897 | 21136 | 30151 |

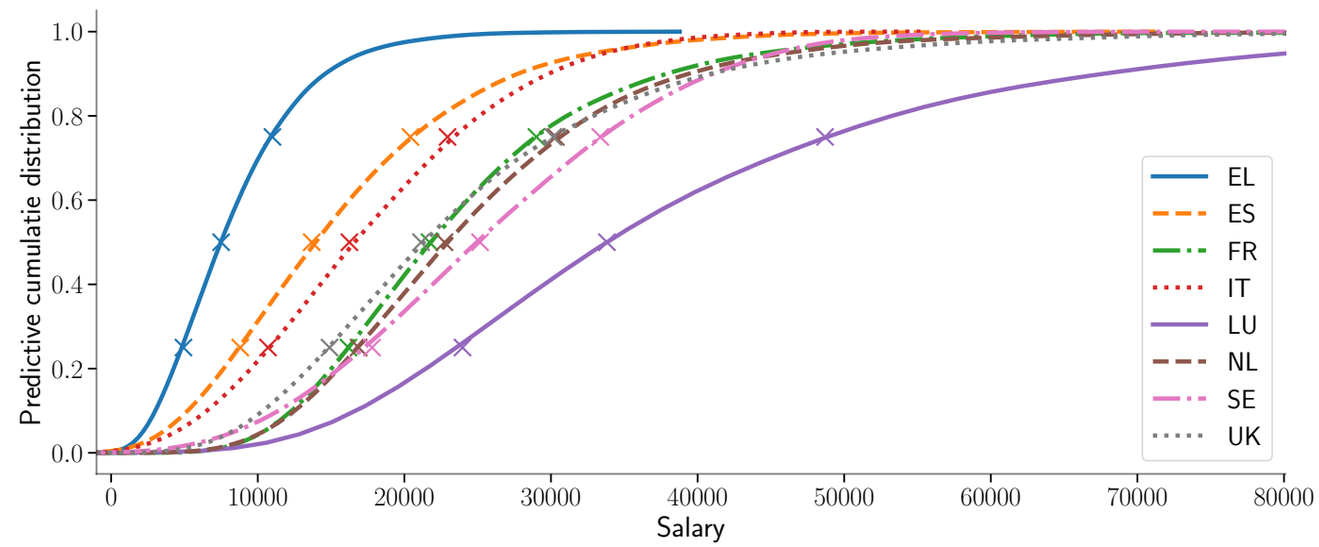
https://ec.europa.eu/eurostat/

Given quantile information, what is the underlying distribution?

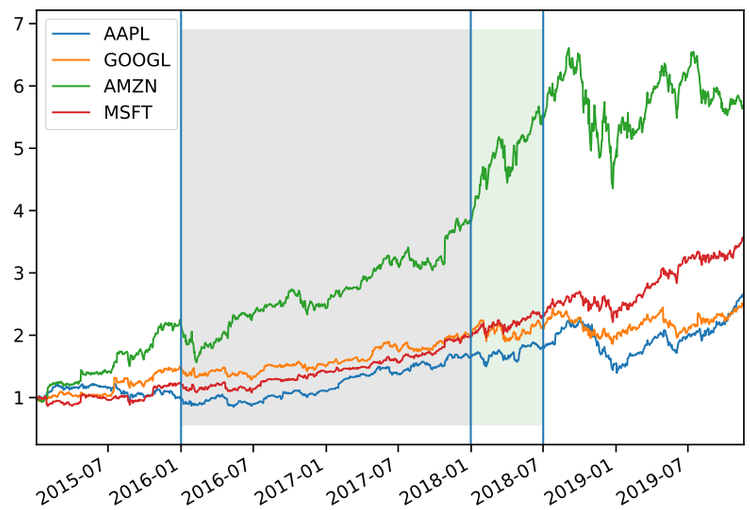
# Bayesian Quantile Matching Estimation

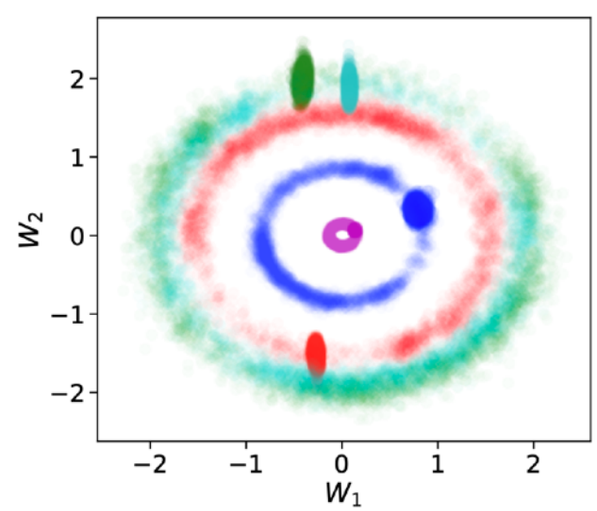
| Country | Sample Size | 25 | 50 | 75 |
|---|---|---|---|---|
| EL | 12918 | 4930 | 7500 | 11000 |
| ES | 19177 | 8803 | 13681 | 20413 |
| FR | 21325 | 16185 | 21713 | 29008 |
| IT | 24969 | 10699 | 16247 | 22944 |
| LU | 10292 | 23964 | 33818 | 48692 |
| NL | 12748 | 16879 | 22733 | 30327 |
| SE | 11635 | 17794 | 25164 | 33365 |
| UK | 17645 | 14897 | 21136 | 30151 |

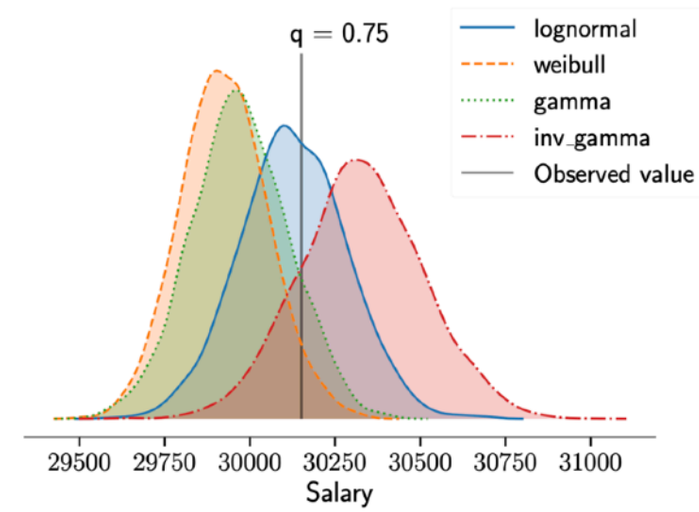
https://ec.europa.eu/eurostat/




| COUNTRY | WEIBULL | LOGNORMAL | GAMMA | INV_GAMMA | FRECHET | CHI_SQUARE |
|---|---|---|---|---|---|---|
| EL | $-6.9\ ^{+0.9}_{-2.1}$ | $4.3\ ^{+0.9}_{-2.0}$ | $\mathbf{10.2}\ ^{+1.0}_{-2.0}$ | $-31.5\ ^{+0.9}_{-2.0}$ | $-81.1\ ^{+1.0}_{-2.2}$ | $-2063.9\ ^{+0.5}_{-1.3}$ |
| ES | $-13.4\ ^{+1.0}_{-1.9}$ | $-0.2\ ^{+1.0}_{-2.1}$ | $\mathbf{10.1}\ ^{+0.9}_{-1.8}$ | $-58.9\ ^{+1.0}_{-2.0}$ | $-130.4\ ^{+1.0}_{-2.1}$ | $-2776.2\ ^{+0.5}_{-1.4}$ |
| FR | $-57.7\ ^{+1.0}_{-2.0}$ | $\mathbf{13.0}\ ^{+0.9}_{-2.0}$ | $3.5\ ^{+1.0}_{-2.2}$ | $-4.4\ ^{+1.0}_{-2.1}$ | $-76.8\ ^{+0.9}_{-2.0}$ | $-5847.6\ ^{+0.5}_{-1.4}$ |
| IT | $\mathbf{9.1}\ ^{+0.9}_{-2.0}$ | $-48.9\ ^{+0.9}_{-2.0}$ | $5.3\ ^{+0.9}_{-2.0}$ | $-155.2\ ^{+0.9}_{-2.0}$ | $-290.0\ ^{+1.0}_{-2.1}$ | $-4500.3\ ^{+0.5}_{-1.4}$ |
| LU | $-47.7\ ^{+1.0}_{-2.0}$ | $\mathbf{9.3}\ ^{+1.0}_{-2.1}$ | $-9.3\ ^{+0.9}_{-2.1}$ | $9.1\ ^{+0.9}_{-2.0}$ | $-10.9\ ^{+0.9}_{-1.9}$ | $-2062.3\ ^{+0.5}_{-1.4}$ |
| NL | $-23.3\ ^{+0.9}_{-1.9}$ | $\mathbf{11.6}\ ^{+1.0}_{-2.0}$ | $9.0\ ^{+0.9}_{-1.9}$ | $-1.9\ ^{+1.0}_{-2.0}$ | $-49.4\ ^{+1.0}_{-2.1}$ | $-3473.8\ ^{+0.5}_{-1.4}$ |
| SE | $\mathbf{11.4}\ ^{+1.0}_{-2.0}$ | $-21.2\ ^{+0.9}_{-2.0}$ | $3.9\ ^{+1.0}_{-2.0}$ | $-63.0\ ^{+1.0}_{-2.0}$ | $-138.7\ ^{+0.9}_{-2.0}$ | $-2910.8\ ^{+0.5}_{-1.3}$ |
| UK | $-62.8\ ^{+0.9}_{-2.0}$ | $\mathbf{12.1}\ ^{+1.0}_{-2.0}$ | $-8.1\ ^{+1.0}_{-2.0}$ | $0.5\ ^{+0.9}_{-1.9}$ | $-45.6\ ^{+0.9}_{-1.9}$ | $-3582.7\ ^{+0.5}_{-1.3}$ |




- Applications of Gaussian process latent variable models in Finance (Intelligent Systems and Applications, 2019)

- Rotation invariant householder parameterization for Bayesian PCA (International Conference on Machine Learning - ICML, 2019)

- Bayesian quantile matching estimation (Preprint, 2020)

# Backup

# Learning/Inference in GPs

Bayes Theorem

$$p(f \mid y) = \frac{p(y \mid f)p(f)}{p(y)}$$

Prior

$$\begin{pmatrix} f(x_1) \\ f(x_2) \\ \vdots \\ f(x_N) \end{pmatrix} \sim \mathcal{N}\left( \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \begin{pmatrix} k_{11} & k_{12} & \cdots & k_{1N} \\ k_{21} & k_{22} & \cdots & k_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ k_{N1} & k_{N2} & \cdots & k_{NN} \end{pmatrix} \right)$$

Posterior

$$\boldsymbol{f}_* \mid \boldsymbol{y} \sim \mathcal{N}(\boldsymbol{\mu}', \boldsymbol{K}')$$

$$\boldsymbol{\mu}' = \boldsymbol{K}_*(\boldsymbol{K} + \sigma^2 \boldsymbol{I})^{-1}\boldsymbol{y}$$

$$\boldsymbol{K}' = \boldsymbol{K}_{**} - \boldsymbol{K}_*(\boldsymbol{K} + \sigma^2 \boldsymbol{I})^{-1}\boldsymbol{K}_*^T$$

Likelihood

$$y_i = f(x_i) + \epsilon$$

$$y_i \mid f_i \sim \mathcal{N}(f_i, \sigma^2)$$

$$\boldsymbol{y} \mid \boldsymbol{f} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{K} + \sigma^2 \boldsymbol{I})$$

Marginal Likelihood

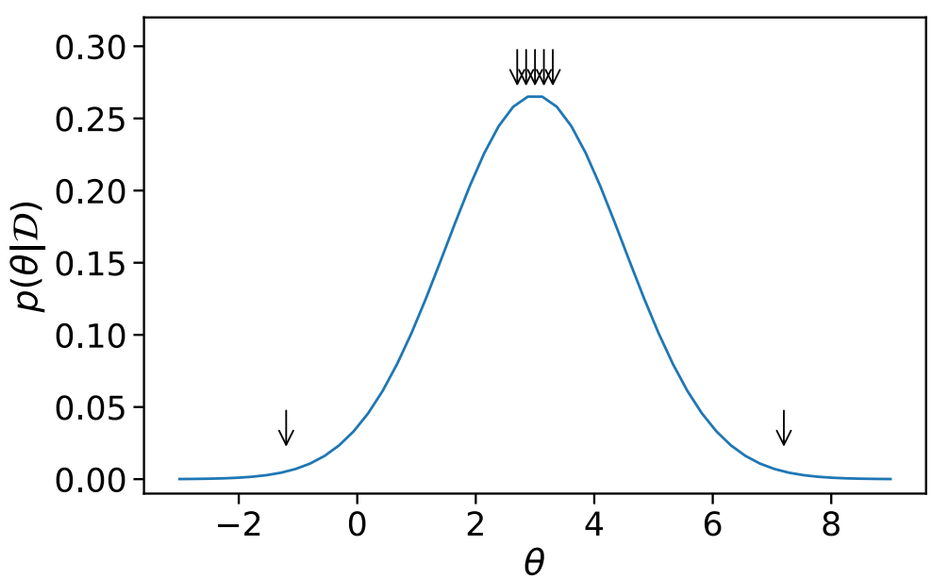
$$\begin{aligned} p(\boldsymbol{y}) &= \int p(\boldsymbol{y} \mid \boldsymbol{f})p(\boldsymbol{f})d\boldsymbol{f} \\ &= -\frac{1}{2}\boldsymbol{y}^T(\boldsymbol{K} + \sigma^2 \boldsymbol{I})^{-1}\boldsymbol{y} \\ &\quad -\frac{1}{2}\log|\boldsymbol{K} + \sigma^2 \boldsymbol{I}| - \frac{N}{2}\log 2\pi \end{aligned}$$

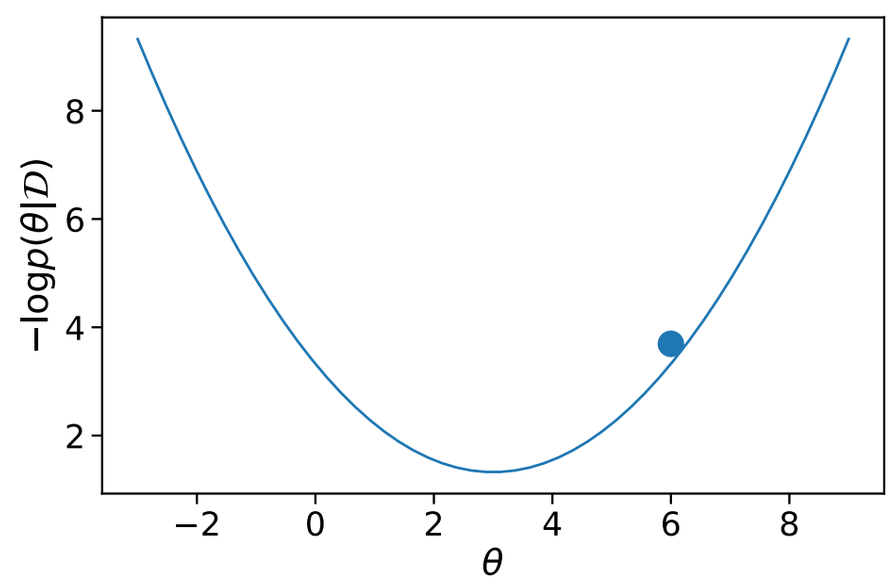
# Sampling (HMC)

- Approximate $p(\boldsymbol{\theta} \mid \mathscr{D})$ by $N$ samples

$$\mathbb{E}_p[f] = \int p(\boldsymbol{\theta} \mid \mathscr{D})\, f(\boldsymbol{\theta})\, d\boldsymbol{\theta} \approx \frac{1}{N} \sum_{n=1}^{N} f(\boldsymbol{\theta}_n)$$

- HMC: Particle moving in probability landscape according to Hamiltons equations with random Gaussian “kicks”



# Sampling (HMC)

$\mathbf{H}(\boldsymbol{\theta}, \boldsymbol{p}) = \mathbf{U}(\boldsymbol{\theta}) + \mathbf{K}(\boldsymbol{p}) = \textbf{const}$

$\mathbf{U}(\boldsymbol{\theta}) = -\log p(\boldsymbol{\theta} \,|\, \mathscr{D})$ $\qquad$ $\mathbf{K}(\boldsymbol{p}) = \dfrac{\boldsymbol{p}^2}{2m}$

$p(\boldsymbol{\theta}, \boldsymbol{p}) \propto \exp(-\mathbf{H}) = p(\boldsymbol{\theta} \,|\, \mathscr{D})\ e^{-\boldsymbol{p}^2/2m}$

$\rightarrow$ $\boldsymbol{\theta}$ and $\boldsymbol{p}$ are uncorrelated

$\rightarrow$ $\boldsymbol{p}$ is Gaussian

**Hamilton's Equations:**

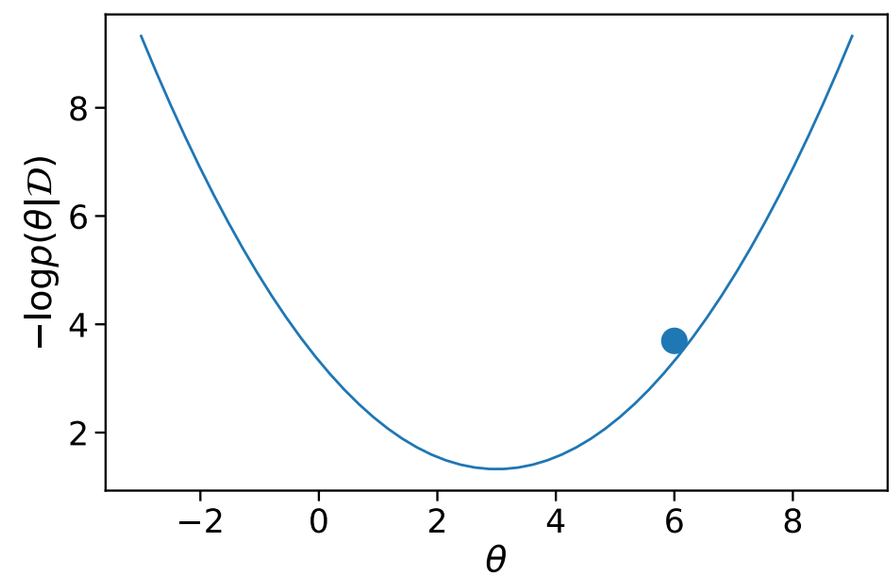
$$\dot{\boldsymbol{\theta}} = \frac{\partial \mathbf{H}}{\partial \boldsymbol{p}} \qquad \dot{\boldsymbol{p}} = -\frac{\partial \mathbf{H}}{\partial \boldsymbol{\theta}}$$

https://github.com/PyMLVizard/PyMLViz

**Sampling (in Theory):**

1. Choose some $\boldsymbol{\theta}$ at random
2. N times:
   1. $\boldsymbol{p} \sim \mathscr{N}(0,1)$
   2. Solve for path with $\mathrm{H}(\boldsymbol{\theta}, \boldsymbol{p})$ for fixed amount of time
   3. Save $\boldsymbol{\theta}$ and $\boldsymbol{p}$ as a sample

Chain of $\boldsymbol{\theta}$s converge to $p(\boldsymbol{\theta} \,|\, \mathscr{D})$

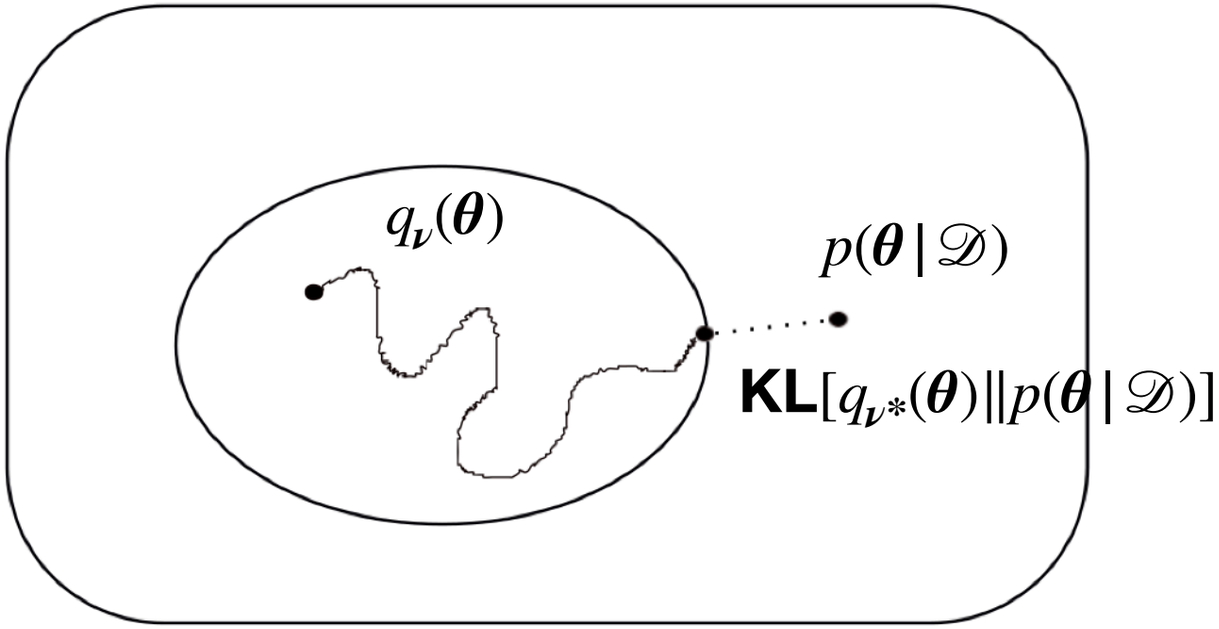


# Variational Bayes

- Approximate $p(\boldsymbol{\theta} \mid \mathscr{D})$ by $q_{\boldsymbol{\nu}}(\boldsymbol{\theta})$
- Goodness is measured in e.g. Kullback-Leibler divergence

$$\mathbf{KL}[q_{\boldsymbol{\nu}}(\boldsymbol{\theta}) \| p(\boldsymbol{\theta} \mid \mathscr{D})] = \int q_{\boldsymbol{\nu}}(\boldsymbol{\theta}) \ln \frac{q_{\boldsymbol{\nu}}(\boldsymbol{\theta})}{p(\boldsymbol{\theta} \mid \mathscr{D})} d\boldsymbol{\theta}$$

**Turns Inference Into Optimization**

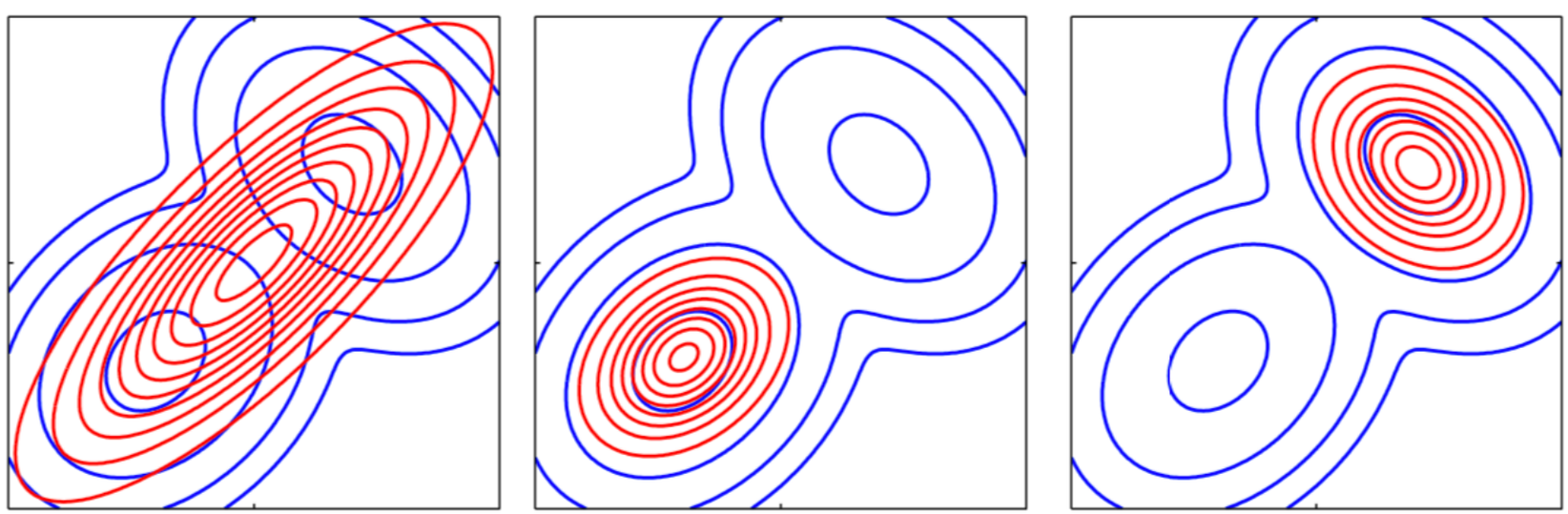


# Variational Bayes

$$\ln p(\mathscr{D}) = \textbf{ELBO} + \textbf{KL}[q_{\boldsymbol{\nu}}(\boldsymbol{\theta}) \| p(\boldsymbol{\theta} \,|\, \mathscr{D})]$$

$$\textbf{ELBO} = \int q_{\boldsymbol{\nu}}(\boldsymbol{\theta}) \ln \frac{p(\mathscr{D}) \,|\, \boldsymbol{\theta}) p(\boldsymbol{\theta})}{q_{\boldsymbol{\nu}}(\boldsymbol{\theta})} d\boldsymbol{\theta} = -\int q_{\boldsymbol{\nu}}(\boldsymbol{\theta}) \ln \frac{q_{\boldsymbol{\nu}}(\boldsymbol{\theta})}{\tilde{p}(\boldsymbol{\theta} \,|\, \mathscr{D})} d\boldsymbol{\theta}$$

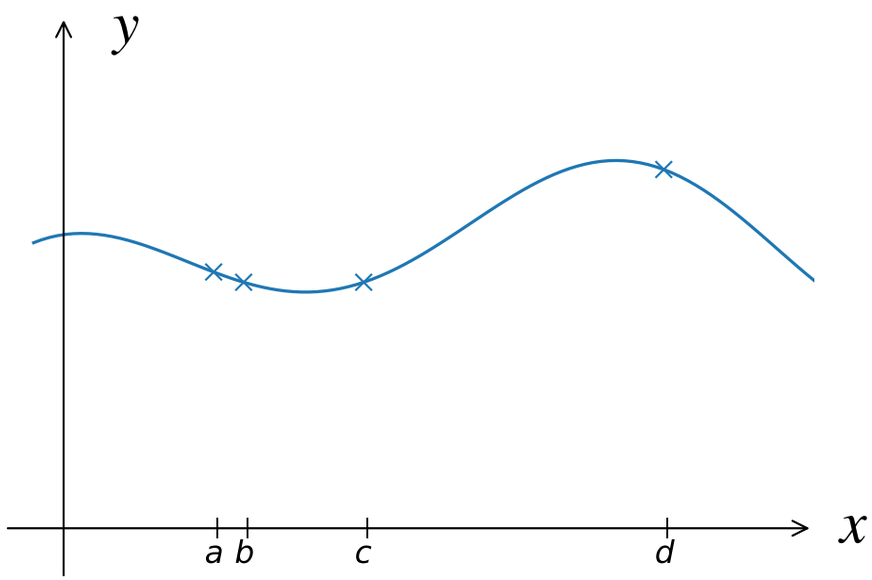
$$\textbf{KL}[q_{\boldsymbol{\nu}}(\boldsymbol{\theta}) \| p(\boldsymbol{\theta} \,|\, \mathscr{D})] = \int q_{\boldsymbol{\nu}}(\boldsymbol{\theta}) \ln \frac{q_{\boldsymbol{\nu}}(\boldsymbol{\theta})}{p(\boldsymbol{\theta} \,|\, \mathscr{D})} d\boldsymbol{\theta}$$



*Pattern Recognition and Machine Learning, Christopher M. Bishop*

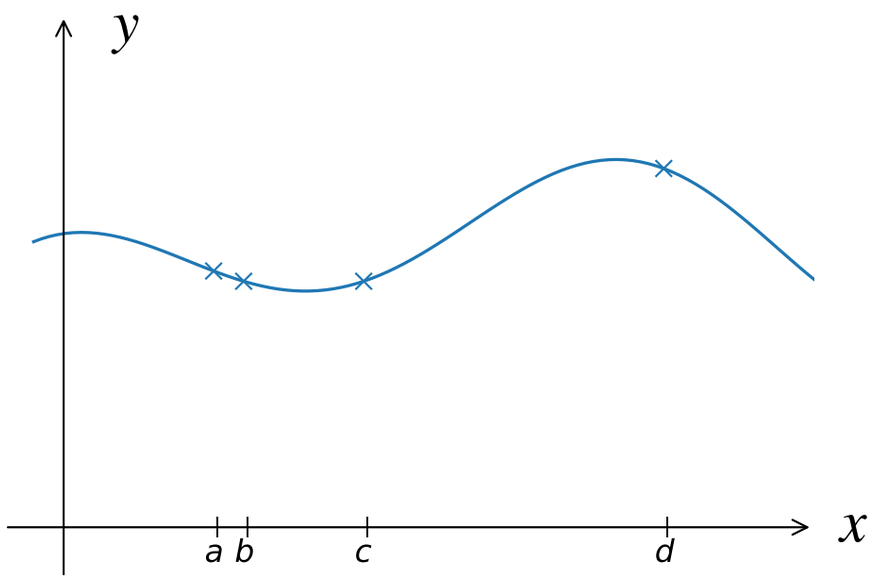
# Latent Variable Models

$$\boldsymbol{X} \in \mathbb{R}^{N \times Q} \quad \overset{f}{\rightarrow} \quad \boldsymbol{Y} \in \mathbb{R}^{N \times D}$$



# Latent Variable Models

$$\boldsymbol{X} \in \mathbb{R}^{N \times Q} \quad \overset{f}{\rightarrow} \quad \boldsymbol{Y} \in \mathbb{R}^{N \times D}$$

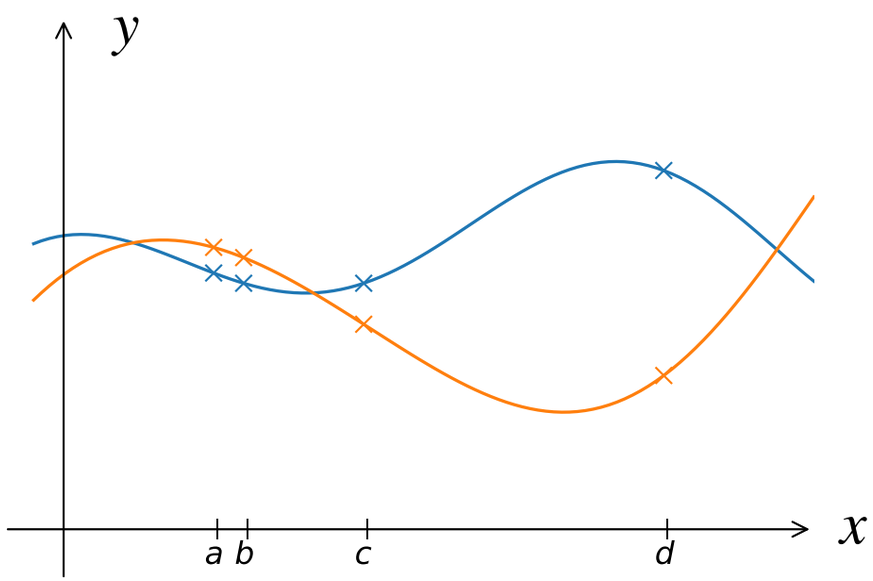



$$\boldsymbol{X} \quad \rightarrow \quad \boldsymbol{Y}$$

$$\begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} \xrightarrow{f_1} \begin{pmatrix} a_1 & & \\ b_1 & & \\ c_1 & & \\ d_1 & & \end{pmatrix}$$

# Latent Variable Models

$$\boldsymbol{X} \in \mathbb{R}^{N \times Q} \quad \overset{f}{\rightarrow} \quad \boldsymbol{Y} \in \mathbb{R}^{N \times D}$$

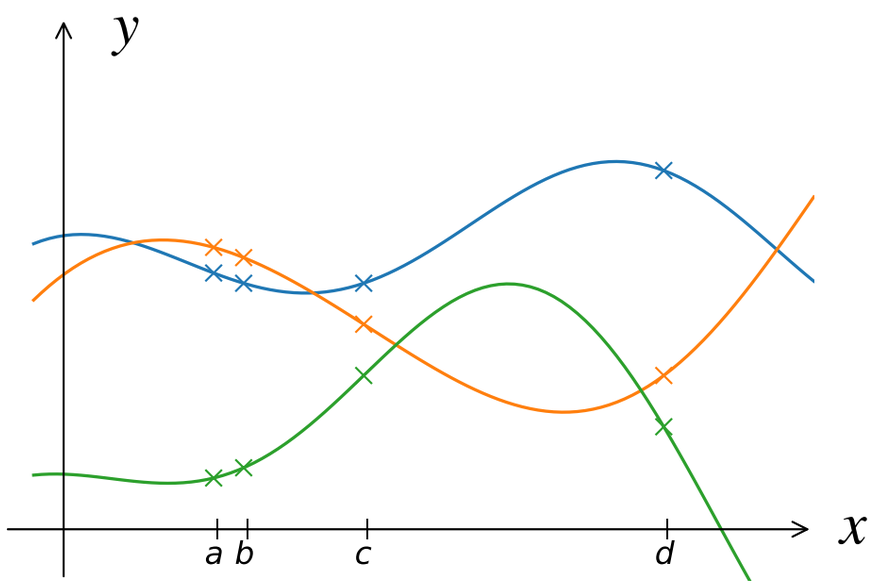



$$\boldsymbol{X} \quad \rightarrow \quad \boldsymbol{Y}$$

$$\begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} \xrightarrow{f_2} \begin{pmatrix} a_1 & a_2 \\ b_1 & b_2 \\ c_1 & c_2 \\ d_1 & d_2 \end{pmatrix}$$

# Latent Variable Models

$$\boldsymbol{X} \in \mathbb{R}^{N \times Q} \quad \overset{f}{\rightarrow} \quad \boldsymbol{Y} \in \mathbb{R}^{N \times D}$$

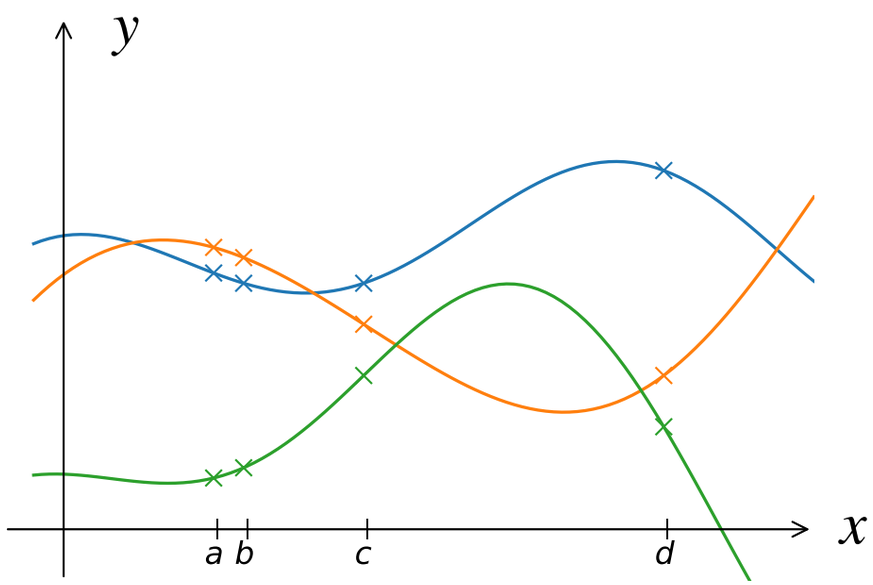



$$\boldsymbol{X} \quad \rightarrow \quad \boldsymbol{Y}$$

$$\begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} \xrightarrow{f_3} \begin{pmatrix} a_1 & a_2 & a_3 & \\ b_1 & b_2 & b_3 & \dots \\ c_1 & c_2 & c_3 & \\ d_1 & d_2 & d_3 & \end{pmatrix}$$

# Latent Variable Models

$$\boldsymbol{X} \in \mathbb{R}^{N \times Q} \quad \overset{f}{\rightarrow} \quad \boldsymbol{Y} \in \mathbb{R}^{N \times D}$$




$$\boldsymbol{X} \quad \rightarrow \quad \boldsymbol{Y}$$

$$\begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} \xrightarrow{f_i} \begin{pmatrix} a_1 & a_2 & a_3 & \\ b_1 & b_2 & b_3 & \dots \\ c_1 & c_2 & c_3 & \\ d_1 & d_2 & d_3 & \end{pmatrix}$$

- Can we infer the hidden state $\boldsymbol{X}$ using only $\boldsymbol{Y}$ ? Yes
- Inference using GPs also gives us the covariance $\boldsymbol{K}$ between different points

# GPLVM as extension to APT

Capital Assert Pricing Model (CAPM)

$$\tilde{r}_n = \alpha_n + r_f + \beta_n(\tilde{r}_m - r_f) + \epsilon_n$$

$$r_n = \tilde{r}_n - r_f \qquad r_m = \tilde{r}_m - r_f$$

$$r_n = r_m \beta_n + \epsilon_n$$

$$r_{n,d} = r_{m,d} \beta_n + \epsilon_{n,d}$$

Arbitrage theory of capital asset pricing (APT)

$$\boldsymbol{r} = \boldsymbol{B}\boldsymbol{F}^T + \boldsymbol{\epsilon}$$

$$\boldsymbol{r} \in \mathbb{R}^{N \times D}$$
$$\boldsymbol{B} \in \mathbb{R}^{N \times Q}$$
$$\boldsymbol{F} \in \mathbb{R}^{D \times Q}$$

$$\boldsymbol{r}_{n,d} = \boldsymbol{F}_{:,d} \boldsymbol{B}_{n,:} + \boldsymbol{\epsilon}_{n,d}$$

$$p(\boldsymbol{r} \,|\, \boldsymbol{B}) = \prod_{d=1}^{D} \mathcal{N}(\boldsymbol{r}_{:,d} \,|\, \boldsymbol{0}, \boldsymbol{B}\boldsymbol{B}^T + \boldsymbol{\Psi})$$

Gaussian process latent variable model (GPLVM)

$$p(\boldsymbol{r} \,|\, \boldsymbol{B}) = \prod_{d=1}^{D} \mathcal{N}(\boldsymbol{r}_{:,d} \,|\, \boldsymbol{0}, \boldsymbol{K}(\boldsymbol{B}, \boldsymbol{B}) + \boldsymbol{\Psi})$$

$$\boldsymbol{r}_{n,d} = f_d(\boldsymbol{B}_{n,:}) + \boldsymbol{\epsilon}_{n,d}$$

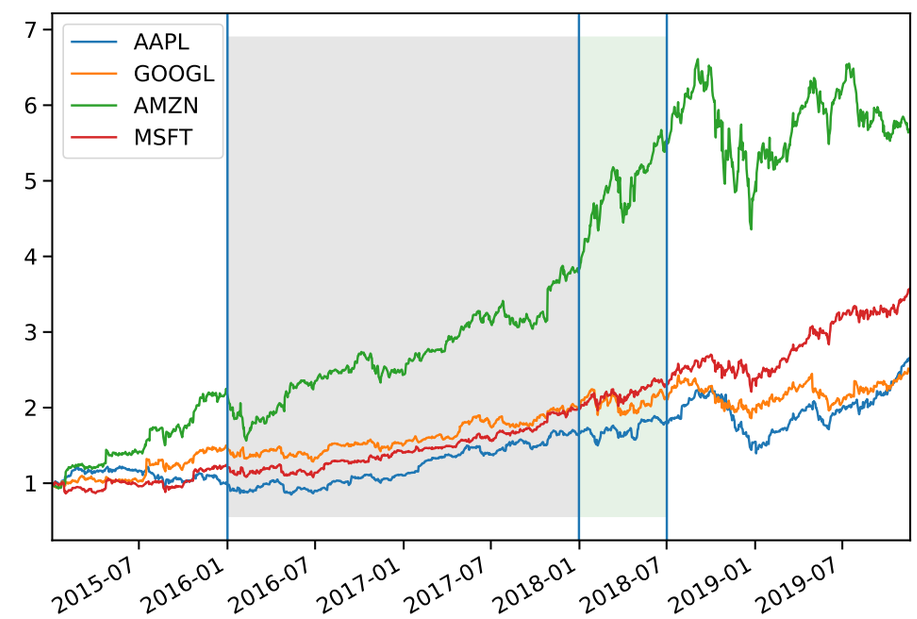
# Portfolio Allocation

Given $N$ stocks, how should I weight them to get an optimal portfolio?

**Markowitz Portfolio Theory**

$$\boldsymbol{w}_{opt} = \min_{\boldsymbol{w}} \left( \boldsymbol{w}^T \boldsymbol{K} \boldsymbol{w} - q \boldsymbol{w}^T \boldsymbol{\mu} \right)$$

**Learn weights on previous 2 years**
**Hold portfolio for next 6 months**



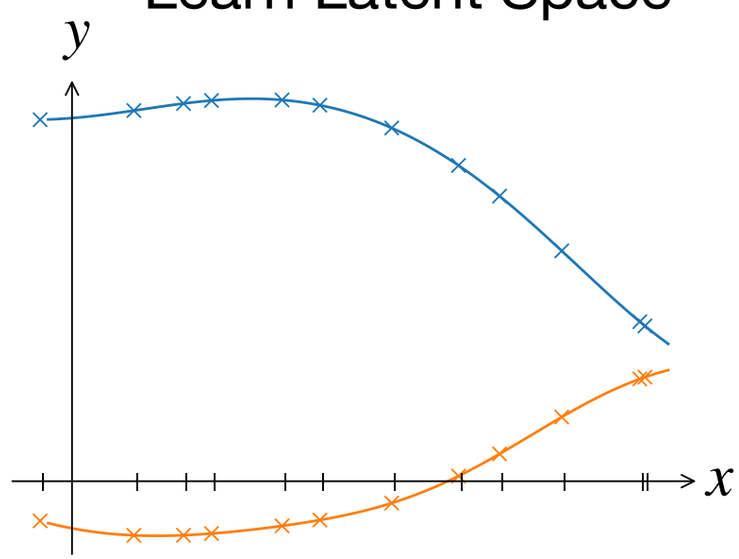

**Backtesting on S&P500 from 2002 to 2018**

| Model | Linear | SE | EXP | M32 | Sample Cov | Ledoit Wolf | Eq. Weighted |
|---|---|---|---|---|---|---|---|
| Mean | 0.142 | 0.151 | 0.155 | 0.158 | 0.149 | 0.148 | 0.182 |
| Std | 0.158 | 0.156 | 0.154 | 0.153 | 0.159 | 0.159 | 0.232 |
| Sharpe ratio | 0.901 | 0.969 | 1.008 | 1.029 | 0.934 | 0.931 | 0.786 |

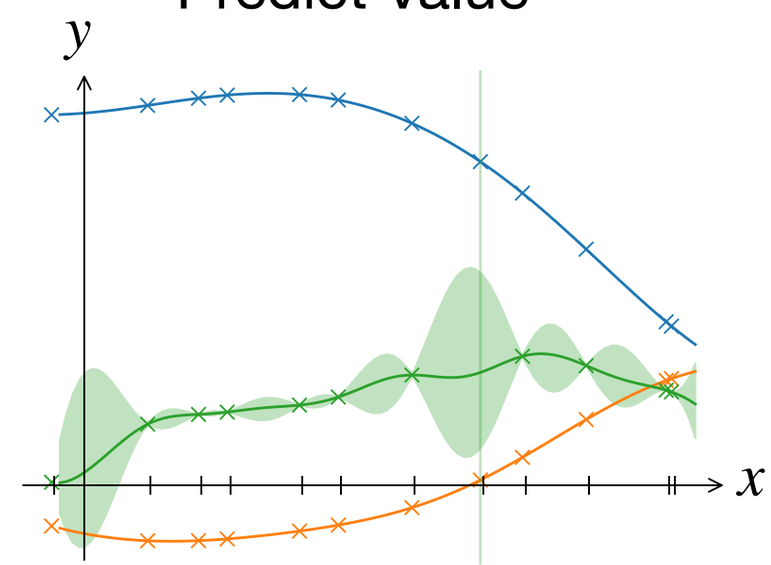
# Predict Missing Values

$$\begin{pmatrix} & a_1 & a_2 & a_3 \\ & b_1 & b_2 & NaN \\ \cdots & c_1 & c_2 & c_3 \\ & d_1 & d_2 & d_3 \end{pmatrix}$$

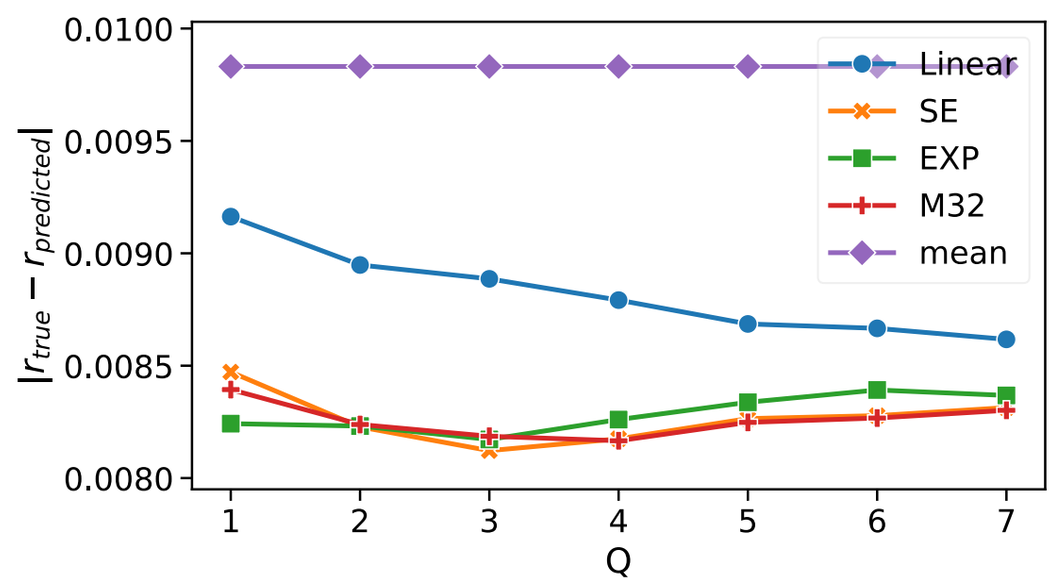
Learn Latent Space




Predict Value




**Prediction of missing values for held out dataset**



# Probabilistic PCA

## Probabilistic PCA (PPCA)

Viewed as a generative model, that maps the latent space $\boldsymbol{X}$ to the data space $\boldsymbol{Y}$

$$\boldsymbol{X} \in \mathbb{R}^{N \times Q} \quad \rightarrow \quad \boldsymbol{Y} \in \mathbb{R}^{N \times D}$$

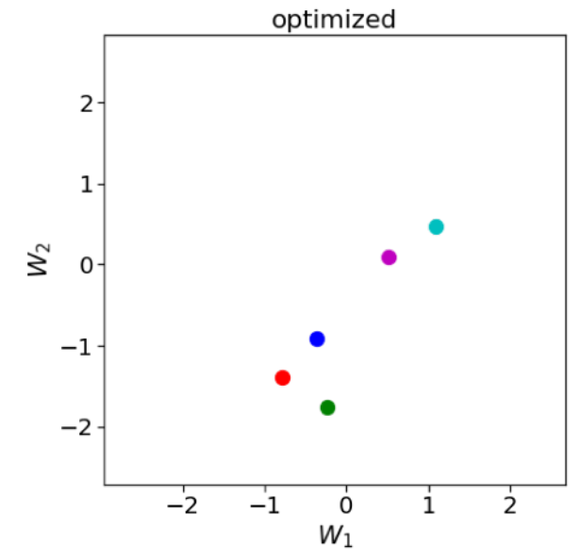
$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{W}^T + \boldsymbol{\epsilon}$$

$$\boldsymbol{X} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}), \quad \epsilon \sim \mathcal{N}\left(\boldsymbol{0}, \sigma^2 \boldsymbol{I}\right)$$

$$p(\boldsymbol{Y} \mid \boldsymbol{W}) = \prod_{n=1}^{N} \mathcal{N}\left(\boldsymbol{Y}_{n,:} \mid \boldsymbol{0}, \boldsymbol{W}\boldsymbol{W}^T + \sigma^2 \boldsymbol{I}\right)$$

$$\boldsymbol{W}\boldsymbol{R}\boldsymbol{R}^T\boldsymbol{W}^T = \boldsymbol{W}\boldsymbol{W}^T \quad \forall\ \boldsymbol{R}\boldsymbol{R}^T = \boldsymbol{I}$$

- Optimization for D=5, Q=2




$$\begin{pmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \\ w_{31} & w_{32} \\ w_{41} & w_{42} \\ w_{51} & w_{52} \end{pmatrix}$$

# Probabilistic PCA

## Probabilistic PCA (PPCA)

Viewed as a generative model, that maps the latent space $\boldsymbol{X}$ to the data space $\boldsymbol{Y}$

$$\boldsymbol{X} \in \mathbb{R}^{N \times Q} \quad \rightarrow \quad \boldsymbol{Y} \in \mathbb{R}^{N \times D}$$

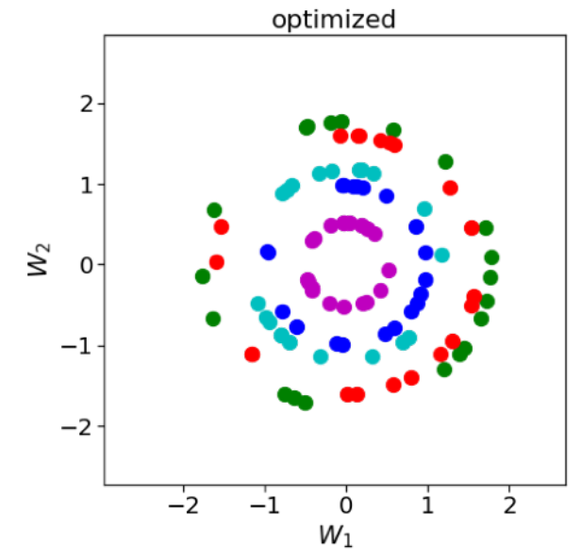
$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{W}^T + \boldsymbol{\epsilon}$$

$$\boldsymbol{X} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}), \quad \epsilon \sim \mathcal{N}\left(\boldsymbol{0}, \sigma^2 \boldsymbol{I}\right)$$

$$p(\boldsymbol{Y} \mid \boldsymbol{W}) = \prod_{n=1}^{N} \mathcal{N}\left(\boldsymbol{Y}_{n,:} \mid \boldsymbol{0}, \boldsymbol{W}\boldsymbol{W}^T + \sigma^2 \boldsymbol{I}\right)$$

$$\boldsymbol{W}\boldsymbol{R}\boldsymbol{R}^T\boldsymbol{W}^T = \boldsymbol{W}\boldsymbol{W}^T \quad \forall\ \boldsymbol{R}\boldsymbol{R}^T = \boldsymbol{I}$$

- Optimization for D=5, Q=2




$$\begin{pmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \\ w_{31} & w_{32} \\ w_{41} & w_{42} \\ w_{51} & w_{52} \end{pmatrix}$$

# Probabilistic PCA

## Probabilistic PCA (PPCA)

Viewed as a generative model, that maps the latent space $\boldsymbol{X}$ to the data space $\boldsymbol{Y}$

$$\boldsymbol{X} \in \mathbb{R}^{N \times Q} \quad \rightarrow \quad \boldsymbol{Y} \in \mathbb{R}^{N \times D}$$

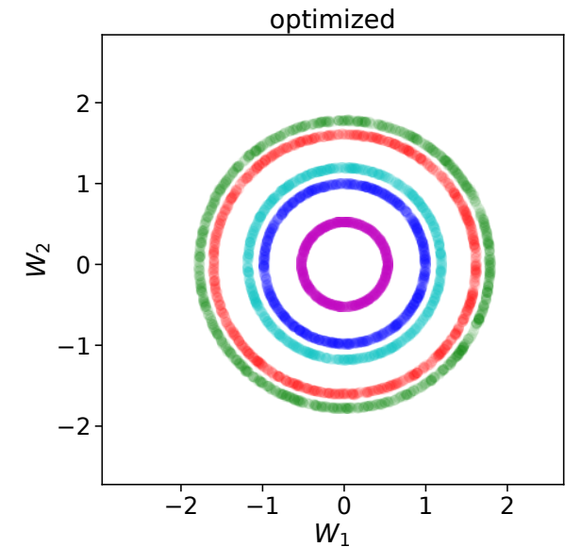
$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{W}^T + \boldsymbol{\epsilon}$$

$$\boldsymbol{X} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}), \quad \epsilon \sim \mathcal{N}\left(\boldsymbol{0}, \sigma^2 \boldsymbol{I}\right)$$

$$p(\boldsymbol{Y} \mid \boldsymbol{W}) = \prod_{n=1}^{N} \mathcal{N}\left(\boldsymbol{Y}_{n,:} \mid \boldsymbol{0}, \boldsymbol{W}\boldsymbol{W}^T + \sigma^2 \boldsymbol{I}\right)$$

$$\boldsymbol{W}\boldsymbol{R}\boldsymbol{R}^T\boldsymbol{W}^T = \boldsymbol{W}\boldsymbol{W}^T \quad \forall\ \boldsymbol{R}\boldsymbol{R}^T = \boldsymbol{I}$$

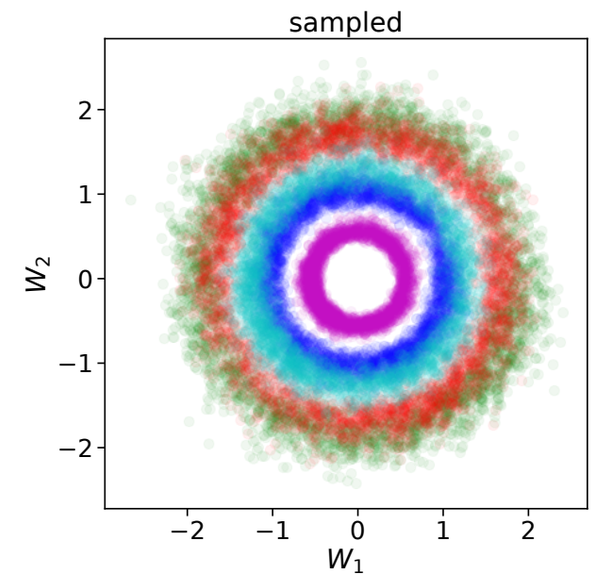
- Rotation invariant likelihood




$$\begin{pmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \\ w_{31} & w_{32} \\ w_{41} & w_{42} \\ w_{51} & w_{52} \end{pmatrix}$$

# Bayesian approach to PPCA

$$p\left(\boldsymbol{W} \mid \boldsymbol{Y}\right) = \frac{p\left(\boldsymbol{Y} \mid \boldsymbol{W}\right) p\left(\boldsymbol{W}\right)}{p(\boldsymbol{Y})}$$




- Sampling is challenging
- Posterior averages are meaningless
- Interpretation of the latent space is almost impossible

$\boldsymbol{U} \sim ?$
$\boldsymbol{\Sigma} \sim ?$ $\rightarrow$ $\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{\Sigma}^T\boldsymbol{U}^T$ **Wishart**

# Theory

- Since $\boldsymbol{U}$, $\boldsymbol{\Sigma}$ is SVD of $\boldsymbol{W}$ → $\boldsymbol{U}$ is a orthogonal matrix

$\boldsymbol{U} \in \mathscr{V}_{Q,D}$ **Stiefel manifold** $\mathscr{V}_{Q,D} = \left\{\boldsymbol{U} \in \mathbb{R}^{D\times Q} \,|\, \boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I}\right\}$

- $\boldsymbol{U}$, $\boldsymbol{\Sigma}^2$ is eigenvalue decomposition of $\boldsymbol{W}\boldsymbol{W}^T$

Eigenvectors of Wishart matrix are distributed uniformly in space of orthogonal matrices ( Blai (2007), Uhlig (1994) )

→ $\boldsymbol{U}$ **is uniformly distributed on the Stiefel manifold**

---

- Distribution of the ordered eigenvalue matrix $\boldsymbol{\Sigma}^2$ of $\boldsymbol{W}\boldsymbol{W}^T$ is (James & Lee (2014))

$$p(\boldsymbol{\lambda}) = ce^{-\frac{1}{2}\sum_{q=1}^{Q}\lambda_q}\prod_{q=1}^{Q}\left(\lambda_q^{\frac{D-Q-1}{2}}\prod_{q'=q+1}^{Q}\left|\lambda_q - \lambda_{q'}\right|\right)$$

$$p\left(\sigma_1, \ldots, \sigma_Q\right) = ce^{-\frac{1}{2}\sum_{q=1}^{Q}\sigma_q^2}\prod_{q=1}^{Q}\left(\sigma_q^{D-Q-1}\prod_{q'=q+1}^{Q}\left|\sigma_q^2 - \sigma_{q'}^2\right|\right)\prod_{q=1}^{Q}2\sigma_q$$

# Implementation

**How to uniformly sample $U$ on $\mathscr{V}_{Q,D}$**

**Theorem 2** *Let $\boldsymbol{v}_D, \boldsymbol{v}_{D-1}, \ldots, \boldsymbol{v}_1$ be uniformly distributed on the unit spheres $\mathbb{S}^{D-1}, \ldots, \mathbb{S}^0$ respectively, where $\mathbb{S}^{n-1}$ is the unit sphere in $\mathbb{R}^n$. Furthermore, let $\boldsymbol{H}_n(\boldsymbol{v}_n)$ be the $n$-th Householder transformation as defined in equation (2.20) The product*

$$\boldsymbol{Q} = \boldsymbol{H}_D(\boldsymbol{v}_D)\boldsymbol{H}_{D-1}(\boldsymbol{v}_{D-1})\ldots\boldsymbol{H}_1(\boldsymbol{v}_1) \qquad (2.21)$$

*is a random orthogonal matrix with distribution given by the Haar measure on $O(D)$.*

**Mezzadri (2007)**

**for** $n = D : 1$

$\boldsymbol{v}_n \sim$ **uniform on** $\mathbb{S}^{n-1}$

$$\boldsymbol{u}_n = \frac{\boldsymbol{v}_n + \operatorname{sgn}(\boldsymbol{v}_{n1}) \parallel \boldsymbol{v}_n \parallel \boldsymbol{e}_1}{\left\| \boldsymbol{v}_n + \operatorname{sgn}(\boldsymbol{v}_{n1}) \parallel \boldsymbol{v}_n \parallel \boldsymbol{e}_1 \right\|}$$

$$\tilde{\boldsymbol{H}}_{\boldsymbol{n}}(\boldsymbol{v}_n) = -\operatorname{sgn}(\boldsymbol{v}_{n1})\left(\boldsymbol{I} - 2\boldsymbol{u}_n\boldsymbol{u}_n^T\right)$$

$$\boldsymbol{H}_n = \begin{pmatrix} \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{0} & \tilde{\boldsymbol{H}}_{\boldsymbol{n}} \end{pmatrix}$$

$$\boldsymbol{U} = \boldsymbol{H}_D(\boldsymbol{v}_D)\boldsymbol{H}_{D-1}(\boldsymbol{v}_{D-1})\ldots\boldsymbol{H}_1(\boldsymbol{v}_1)$$

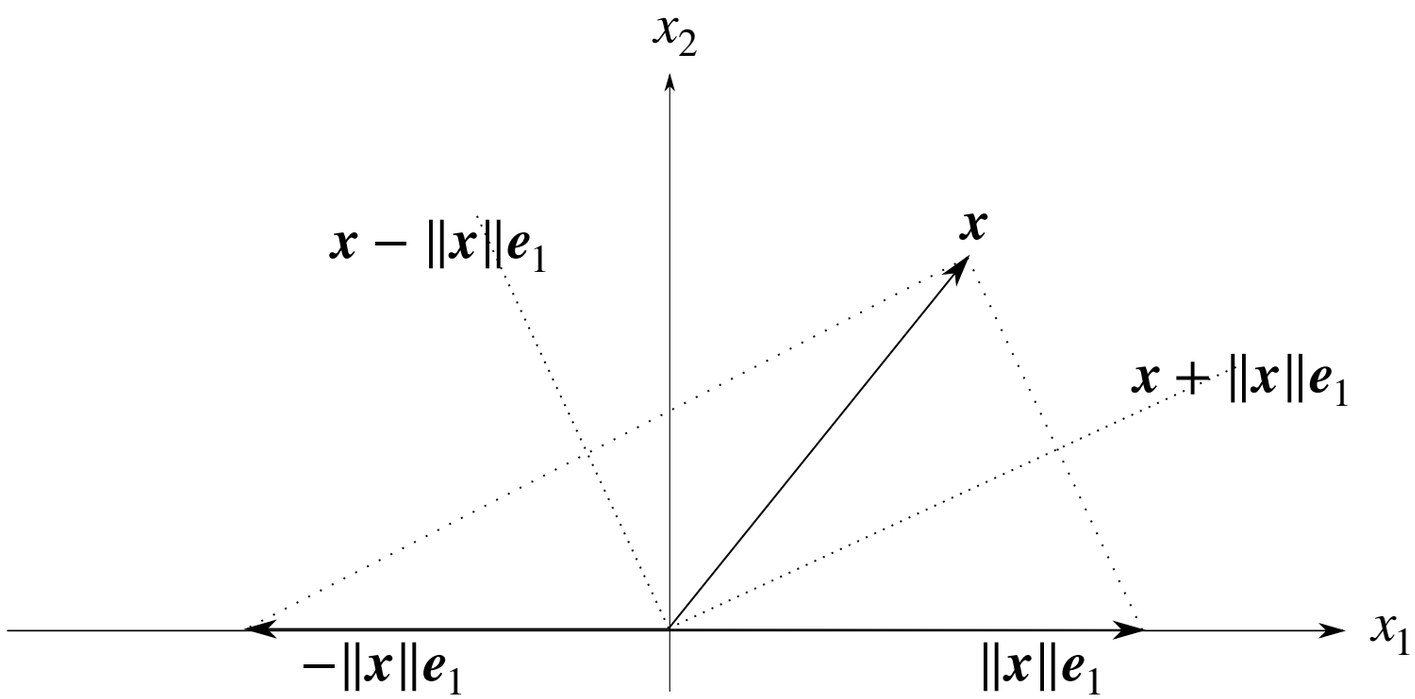
# Implementation

**The full generative model for Bayesian PPCA:**

$$
\begin{aligned}
\boldsymbol{v}_D, \ldots, \boldsymbol{v}_{D-Q+1} &\sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}) \\
\boldsymbol{\sigma} &\sim p(\boldsymbol{\sigma}) \\
\boldsymbol{\mu} &\sim p(\boldsymbol{\mu}) \\
\boldsymbol{U} &= \prod_{q=1}^{Q} \boldsymbol{H}_{D-q+1}\left(\boldsymbol{v}_{D-q+1}\right) \\
\boldsymbol{\Sigma} &= \operatorname{diag}(\boldsymbol{\sigma}) \\
\boldsymbol{W} &= \boldsymbol{U}\boldsymbol{\Sigma} \\
\sigma_{\text{noise}} &\sim p\left(\sigma_{\text{noise}}\right) \\
Y &\sim \prod_{n=1}^{N} \mathcal{N}\left(\boldsymbol{Y}_{n,:} | \boldsymbol{\mu}, \boldsymbol{W}\boldsymbol{W}^T + \sigma_{\text{noise}}^2 \boldsymbol{I}\right)
\end{aligned}
$$

# Householder Transformations

- Used to reflect a vector in such a way that all coordinates but one disappear, e.g.: QR-decomposition




$$\boldsymbol{u} = \boldsymbol{x} \pm \|\boldsymbol{x}\|\boldsymbol{e}_1$$

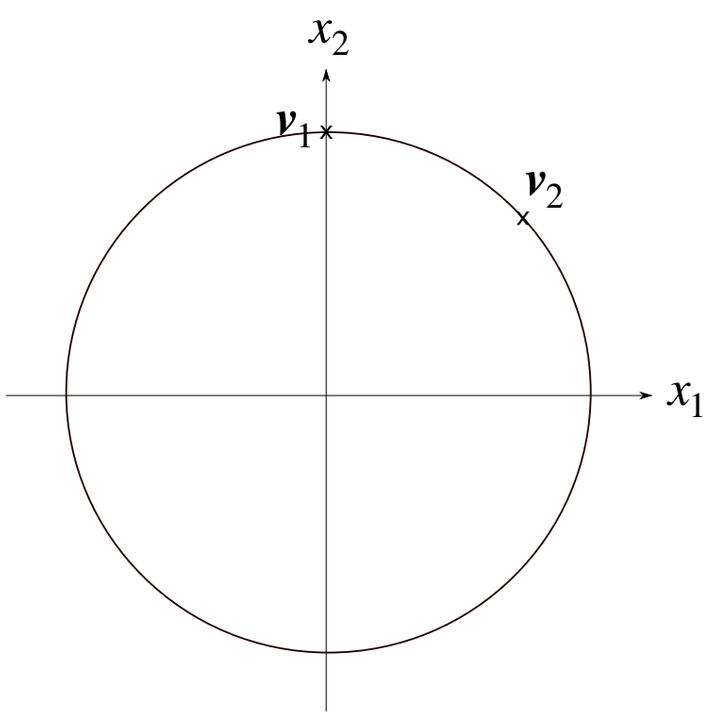
$$\boldsymbol{H} = \boldsymbol{1} - 2\hat{\boldsymbol{u}}\hat{\boldsymbol{u}}^T$$

$$\boldsymbol{H}\boldsymbol{x} = \|\boldsymbol{x}\|\boldsymbol{e}_1$$

# Householder Transformations

Example for $D = 2$

$$\boldsymbol{U} \in \mathscr{V}_{2,2} \qquad \boldsymbol{v}_1 = \begin{pmatrix} 0 \\ v_{11} \end{pmatrix} \quad \boldsymbol{v}_2 = \begin{pmatrix} v_{21} \\ v_{22} \end{pmatrix}$$




**for** $n = D : 1$

$\boldsymbol{v}_n \sim$ **uniform on** $\mathbb{S}^{n-1}$

$$\boldsymbol{u}_n = \frac{\boldsymbol{v}_n + \operatorname{sgn}(\boldsymbol{v}_{n1}) \parallel \boldsymbol{v}_n \parallel \boldsymbol{e}_1}{\left\| \boldsymbol{v}_n + \operatorname{sgn}(\boldsymbol{v}_{n1}) \parallel \boldsymbol{v}_n \parallel \boldsymbol{e}_1 \right\|}$$

$$\tilde{\boldsymbol{H}}_{\boldsymbol{n}}\left(\boldsymbol{v}_n\right) = -\operatorname{sgn}\left(\boldsymbol{v}_{n1}\right)\left(\boldsymbol{I} - 2\boldsymbol{u}_n\boldsymbol{u}_n^T\right)$$

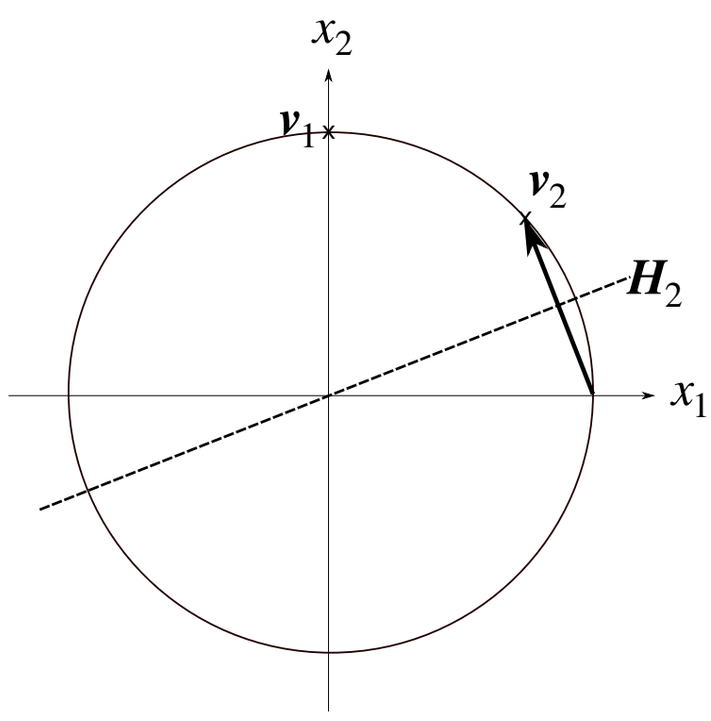
$$\boldsymbol{H}_n = \begin{pmatrix} \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{0} & \tilde{\boldsymbol{H}}_{\boldsymbol{n}} \end{pmatrix}$$

$$\boldsymbol{U} = \boldsymbol{H}_D\left(\boldsymbol{v}_D\right)\boldsymbol{H}_{D-1}\left(\boldsymbol{v}_{D-1}\right)\ldots\boldsymbol{H}_1\left(\boldsymbol{v}_1\right)$$

# Householder Transformations

Example for $D = 2$

$$\boldsymbol{U} \in \mathscr{V}_{2,2} \qquad \boldsymbol{v}_1 = \begin{pmatrix} 0 \\ v_{11} \end{pmatrix} \quad \boldsymbol{v}_2 = \begin{pmatrix} v_{21} \\ v_{22} \end{pmatrix}$$




**for** $n = D : 1$

$\boldsymbol{v}_n \sim$ **uniform on** $\mathbb{S}^{n-1}$

$$\boldsymbol{u}_n = \frac{\boldsymbol{v}_n + \operatorname{sgn}(\boldsymbol{v}_{n1}) \parallel \boldsymbol{v}_n \parallel \boldsymbol{e}_1}{\left\| \boldsymbol{v}_n + \operatorname{sgn}(\boldsymbol{v}_{n1}) \parallel \boldsymbol{v}_n \parallel \boldsymbol{e}_1 \right\|}$$

$$\tilde{\boldsymbol{H}}_{\boldsymbol{n}}(\boldsymbol{v}_n) = -\operatorname{sgn}(\boldsymbol{v}_{n1})\left(\boldsymbol{I} - 2\boldsymbol{u}_n\boldsymbol{u}_n^T\right)$$

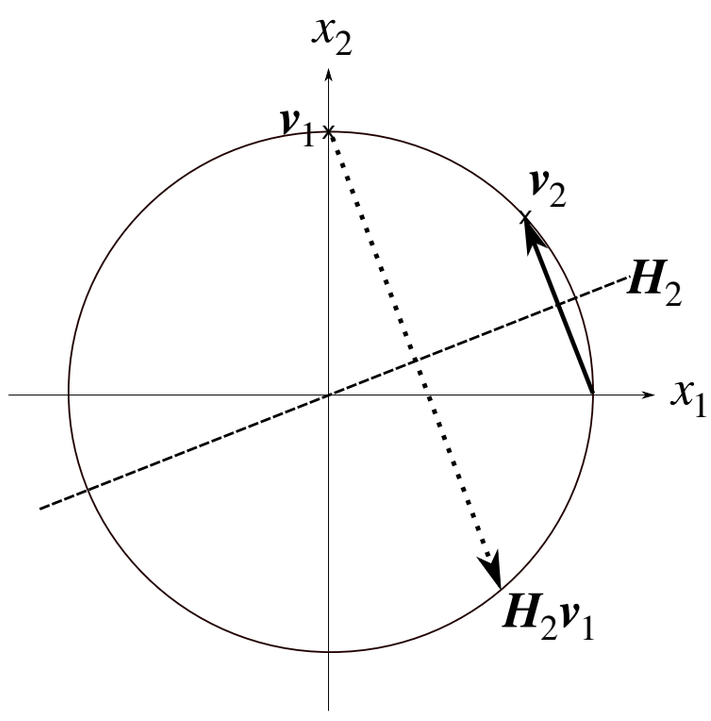
$$\boldsymbol{H}_n = \begin{pmatrix} \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{0} & \tilde{\boldsymbol{H}}_{\boldsymbol{n}} \end{pmatrix}$$

$$\boldsymbol{U} = \boldsymbol{H}_D(\boldsymbol{v}_D)\,\boldsymbol{H}_{D-1}(\boldsymbol{v}_{D-1})\ldots\boldsymbol{H}_1(\boldsymbol{v}_1)$$

# Householder Transformations

Example for $D = 2$

$$\boldsymbol{U} \in \mathscr{V}_{2,2} \qquad \boldsymbol{v}_1 = \begin{pmatrix} 0 \\ v_{11} \end{pmatrix} \quad \boldsymbol{v}_2 = \begin{pmatrix} v_{21} \\ v_{22} \end{pmatrix}$$




**for** $n = D : 1$

$\boldsymbol{v}_n \sim$ **uniform on** $\mathbb{S}^{n-1}$

$$\boldsymbol{u}_n = \frac{\boldsymbol{v}_n + \operatorname{sgn}(\boldsymbol{v}_{n1}) \parallel \boldsymbol{v}_n \parallel \boldsymbol{e}_1}{\left\| \boldsymbol{v}_n + \operatorname{sgn}(\boldsymbol{v}_{n1}) \parallel \boldsymbol{v}_n \parallel \boldsymbol{e}_1 \right\|}$$

$$\tilde{\boldsymbol{H}}_{\boldsymbol{n}}\left(\boldsymbol{v}_n\right) = -\operatorname{sgn}\left(\boldsymbol{v}_{n1}\right)\left(\boldsymbol{I} - 2\boldsymbol{u}_n\boldsymbol{u}_n^T\right)$$

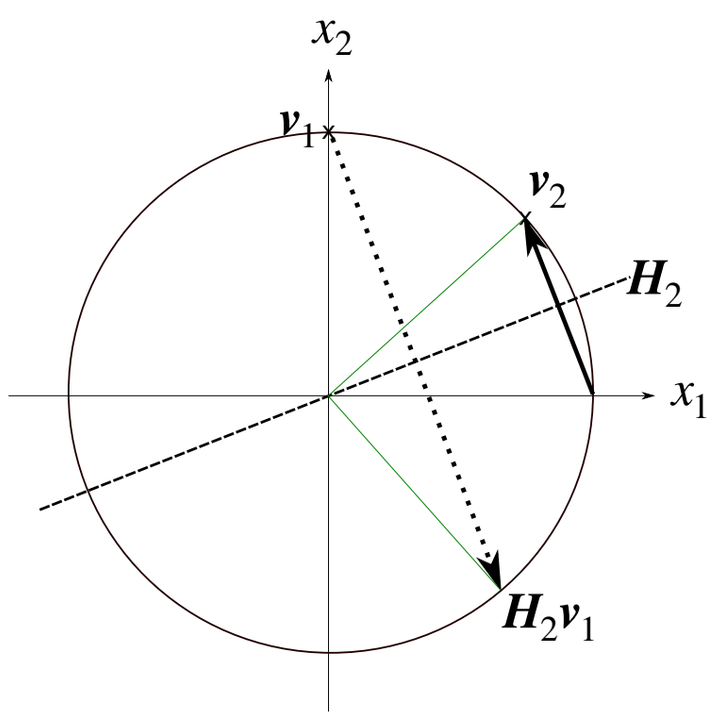
$$\boldsymbol{H}_n = \begin{pmatrix} \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{0} & \tilde{\boldsymbol{H}}_{\boldsymbol{n}} \end{pmatrix}$$

$$\boldsymbol{U} = \boldsymbol{H}_D\left(\boldsymbol{v}_D\right)\boldsymbol{H}_{D-1}\left(\boldsymbol{v}_{D-1}\right)\ldots\boldsymbol{H}_1\left(\boldsymbol{v}_1\right)$$

# Householder Transformations

Example for $D = 2$

$$\boldsymbol{U} \in \mathscr{V}_{2,2} \qquad \boldsymbol{v}_1 = \begin{pmatrix} 0 \\ v_{11} \end{pmatrix} \quad \boldsymbol{v}_2 = \begin{pmatrix} v_{21} \\ v_{22} \end{pmatrix}$$




**for** $n = D : 1$

$\boldsymbol{v}_n \sim$ **uniform on** $\mathbb{S}^{n-1}$

$$\boldsymbol{u}_n = \frac{\boldsymbol{v}_n + \operatorname{sgn}(\boldsymbol{v}_{n1}) \parallel \boldsymbol{v}_n \parallel \boldsymbol{e}_1}{\left\| \boldsymbol{v}_n + \operatorname{sgn}(\boldsymbol{v}_{n1}) \parallel \boldsymbol{v}_n \parallel \boldsymbol{e}_1 \right\|}$$

$$\tilde{\boldsymbol{H}}_{\boldsymbol{n}}\left(\boldsymbol{v}_n\right) = -\operatorname{sgn}\left(\boldsymbol{v}_{n1}\right)\left(\boldsymbol{I} - 2\boldsymbol{u}_n\boldsymbol{u}_n^T\right)$$

$$\boldsymbol{H}_n = \begin{pmatrix} \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{0} & \tilde{\boldsymbol{H}}_{\boldsymbol{n}} \end{pmatrix}$$

$$\boldsymbol{U} = \boldsymbol{H}_D\left(\boldsymbol{v}_D\right)\boldsymbol{H}_{D-1}\left(\boldsymbol{v}_{D-1}\right)\ldots\boldsymbol{H}_1\left(\boldsymbol{v}_1\right)$$

# Householder Transformations

Example for $D = 2$

$$\boldsymbol{U} \in \mathscr{V}_{2,2} \qquad \boldsymbol{v}_1 = \begin{pmatrix} 0 \\ v_{11} \end{pmatrix} \quad \boldsymbol{v}_2 = \begin{pmatrix} v_{21} \\ v_{22} \end{pmatrix}$$

**for** $n = D : 1$

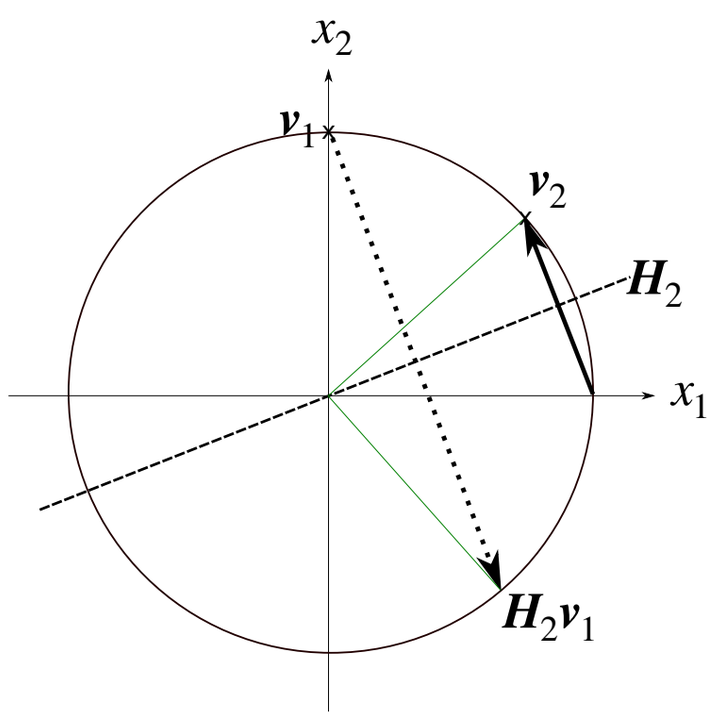
$\boldsymbol{v}_n \sim$ **uniform on** $\mathbb{S}^{n-1}$

$$\boldsymbol{u}_n = \frac{\boldsymbol{v}_n + \operatorname{sgn}(\boldsymbol{v}_{n1}) \parallel \boldsymbol{v}_n \parallel \boldsymbol{e}_1}{\left\| \boldsymbol{v}_n + \operatorname{sgn}(\boldsymbol{v}_{n1}) \parallel \boldsymbol{v}_n \parallel \boldsymbol{e}_1 \right\|}$$

$$\tilde{\boldsymbol{H}}_{\boldsymbol{n}}(\boldsymbol{v}_n) = -\operatorname{sgn}(\boldsymbol{v}_{n1})\left(\boldsymbol{I} - 2\boldsymbol{u}_n\boldsymbol{u}_n^T\right)$$

$$\boldsymbol{H}_n = \begin{pmatrix} \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{0} & \tilde{\boldsymbol{H}}_{\boldsymbol{n}} \end{pmatrix}$$

$$\boldsymbol{U} = \boldsymbol{H}_D(\boldsymbol{v}_D)\boldsymbol{H}_{D-1}(\boldsymbol{v}_{D-1})\ldots\boldsymbol{H}_1(\boldsymbol{v}_1)$$

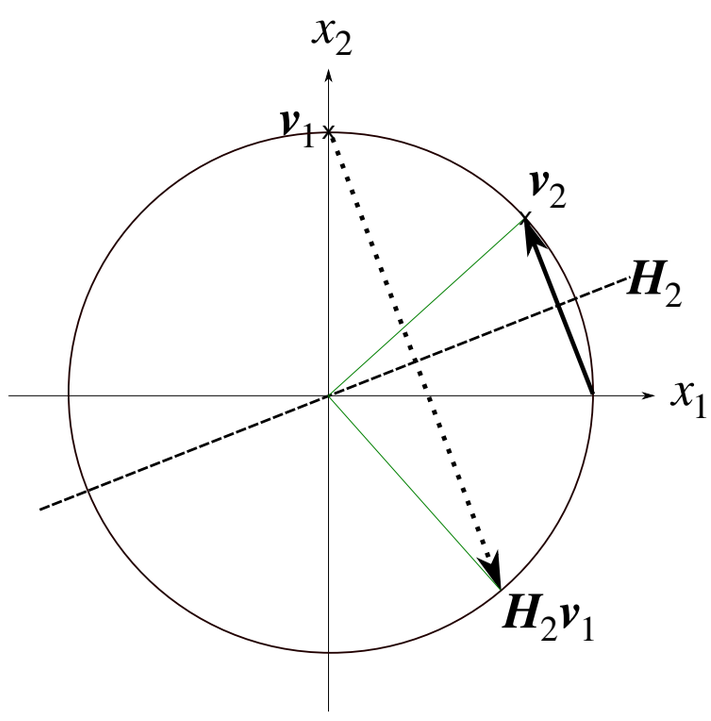



$$\boldsymbol{U} = \boldsymbol{H}_2\boldsymbol{H}_1 = \boldsymbol{H}_2\begin{pmatrix} 1 & 0 \\ 0 & v_{11} \end{pmatrix} = \begin{pmatrix} \boldsymbol{v}_2 & \boldsymbol{H}_2\boldsymbol{v}_1 \end{pmatrix}$$

# Householder Transformations

Example for $D = 2$

$$\boldsymbol{U} \in \mathscr{V}_{2,2} \qquad \boldsymbol{v}_1 = \begin{pmatrix} 0 \\ v_{11} \end{pmatrix} \quad \boldsymbol{v}_2 = \begin{pmatrix} v_{21} \\ v_{22} \end{pmatrix}$$




**for** $n = D : 1$

$\boldsymbol{v}_n \sim$ **uniform on** $\mathbb{S}^{n-1}$

$$\boldsymbol{u}_n = \frac{\boldsymbol{v}_n + \operatorname{sgn}(\boldsymbol{v}_{n1}) \parallel \boldsymbol{v}_n \parallel \boldsymbol{e}_1}{\left\| \boldsymbol{v}_n + \operatorname{sgn}(\boldsymbol{v}_{n1}) \parallel \boldsymbol{v}_n \parallel \boldsymbol{e}_1 \right\|}$$

$$\tilde{\boldsymbol{H}}_{\boldsymbol{n}}\left(\boldsymbol{v}_n\right) = -\operatorname{sgn}\left(\boldsymbol{v}_{n1}\right)\left(\boldsymbol{I} - 2\boldsymbol{u}_n\boldsymbol{u}_n^T\right)$$

$$\boldsymbol{H}_n = \begin{pmatrix} \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{0} & \tilde{\boldsymbol{H}}_{\boldsymbol{n}} \end{pmatrix}$$

$$\boldsymbol{U} = \boldsymbol{H}_D\left(\boldsymbol{v}_D\right)\boldsymbol{H}_{D-1}\left(\boldsymbol{v}_{D-1}\right)\ldots\boldsymbol{H}_1\left(\boldsymbol{v}_1\right)$$

$$\boldsymbol{U} = \boldsymbol{H}_2\boldsymbol{H}_1 = \boldsymbol{H}_2\begin{pmatrix} 1 & 0 \\ 0 & v_{11} \end{pmatrix} = \begin{pmatrix} \boldsymbol{v}_2 & \boldsymbol{H}_2\boldsymbol{v}_1 \end{pmatrix}$$

$$\boldsymbol{U} = \boldsymbol{H}_3\boldsymbol{H}_2\boldsymbol{H}_1 = \begin{pmatrix} \boldsymbol{v}_3 & \boldsymbol{H}_3\boldsymbol{v}_2 & \boldsymbol{H}_3\boldsymbol{H}_2\boldsymbol{v}_1 \end{pmatrix}$$

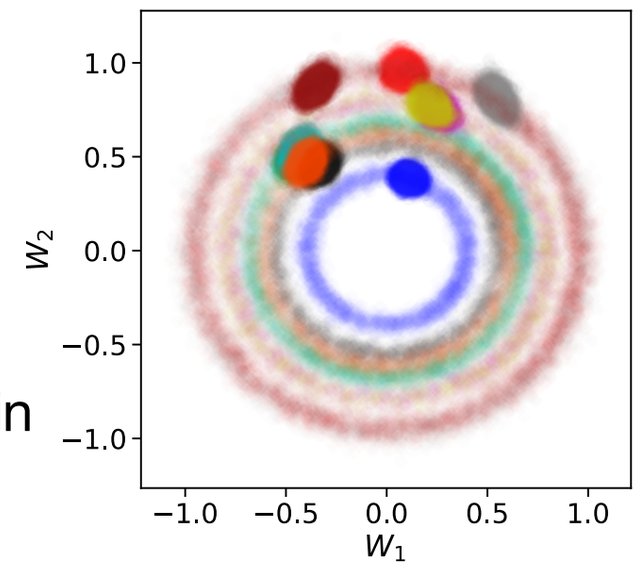
# Results

Breast Cancer

$(N, D) = (569,30)$

Time House: 9.5 min

Time Standard: 25.6 min

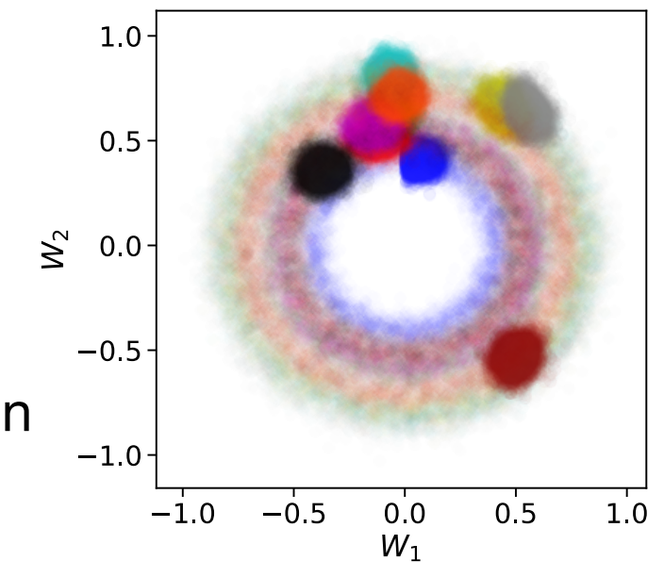



Diabetes

$(N, D) = (442,10)$

Time House: 8.9 min

Time Standard: 2.4 min

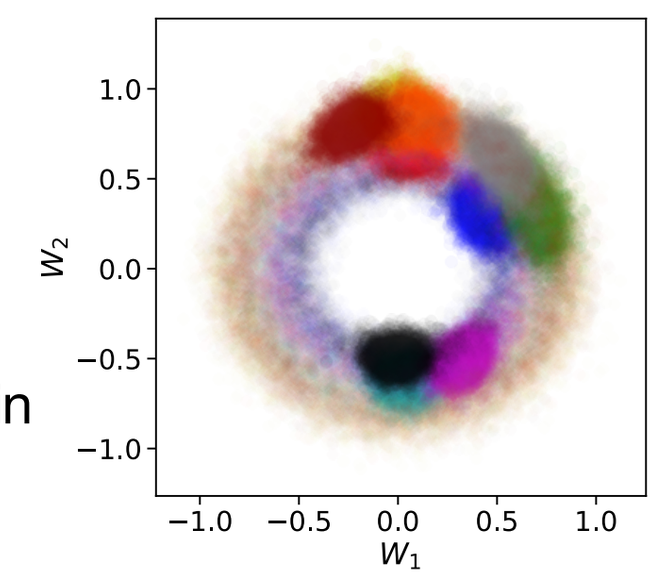



Wine

$(N, D) = (178,13)$

Time House: 0.4 min

Time Standard: 1.2 min

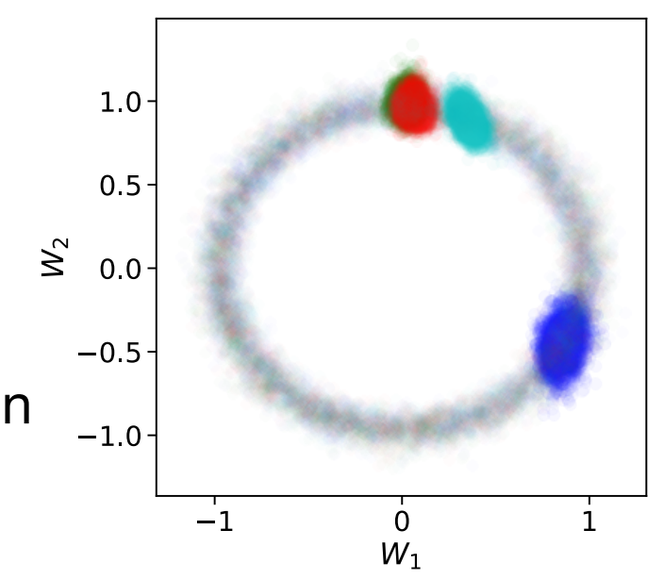



Iris

$(N, D) = (150,4)$

Time House: 0.2 min

Time Standard: 0.8 min



# Results

**for** $n = D : 1$

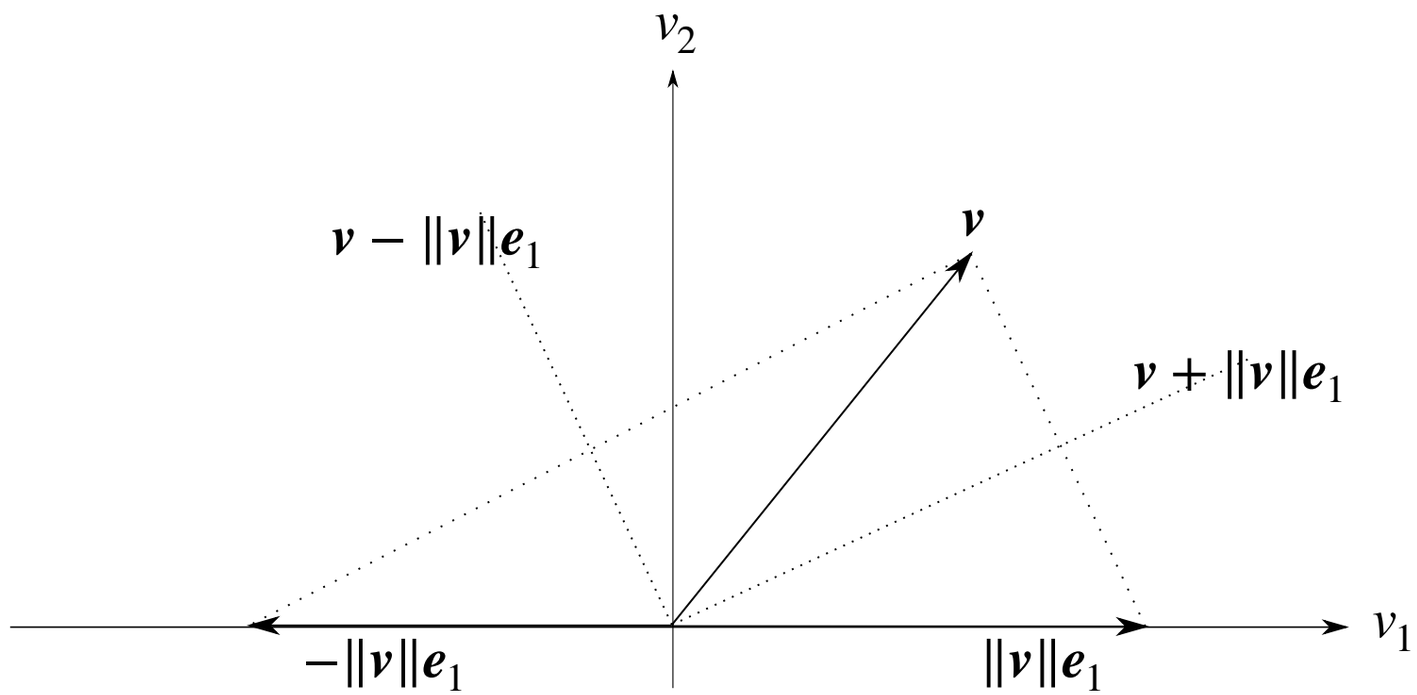
$$\boldsymbol{v}_n \sim \textbf{uniform on } \mathbb{S}^{n-1}$$

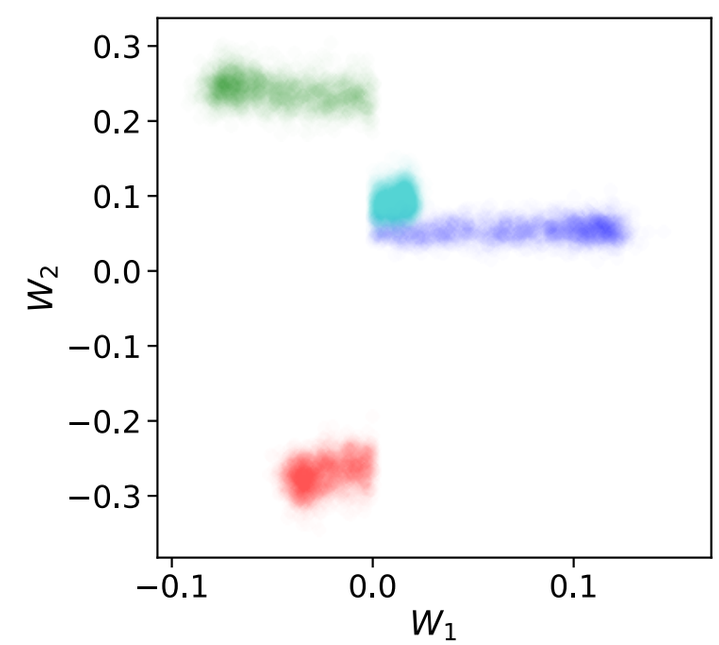
$$\boldsymbol{u}_n = \frac{\boldsymbol{v}_n + \operatorname{sgn}(\boldsymbol{v}_{n1}) \parallel \boldsymbol{v}_n \parallel \boldsymbol{e}_1}{\left\| \boldsymbol{v}_n + \operatorname{sgn}(\boldsymbol{v}_{n1}) \parallel \boldsymbol{v}_n \parallel \boldsymbol{e}_1 \right\|}$$

$$\tilde{\boldsymbol{H}}_{\boldsymbol{n}}(\boldsymbol{v}_n) = -\operatorname{sgn}(\boldsymbol{v}_{n1})\left(\boldsymbol{I} - 2\boldsymbol{u}_n\boldsymbol{u}_n^T\right)$$

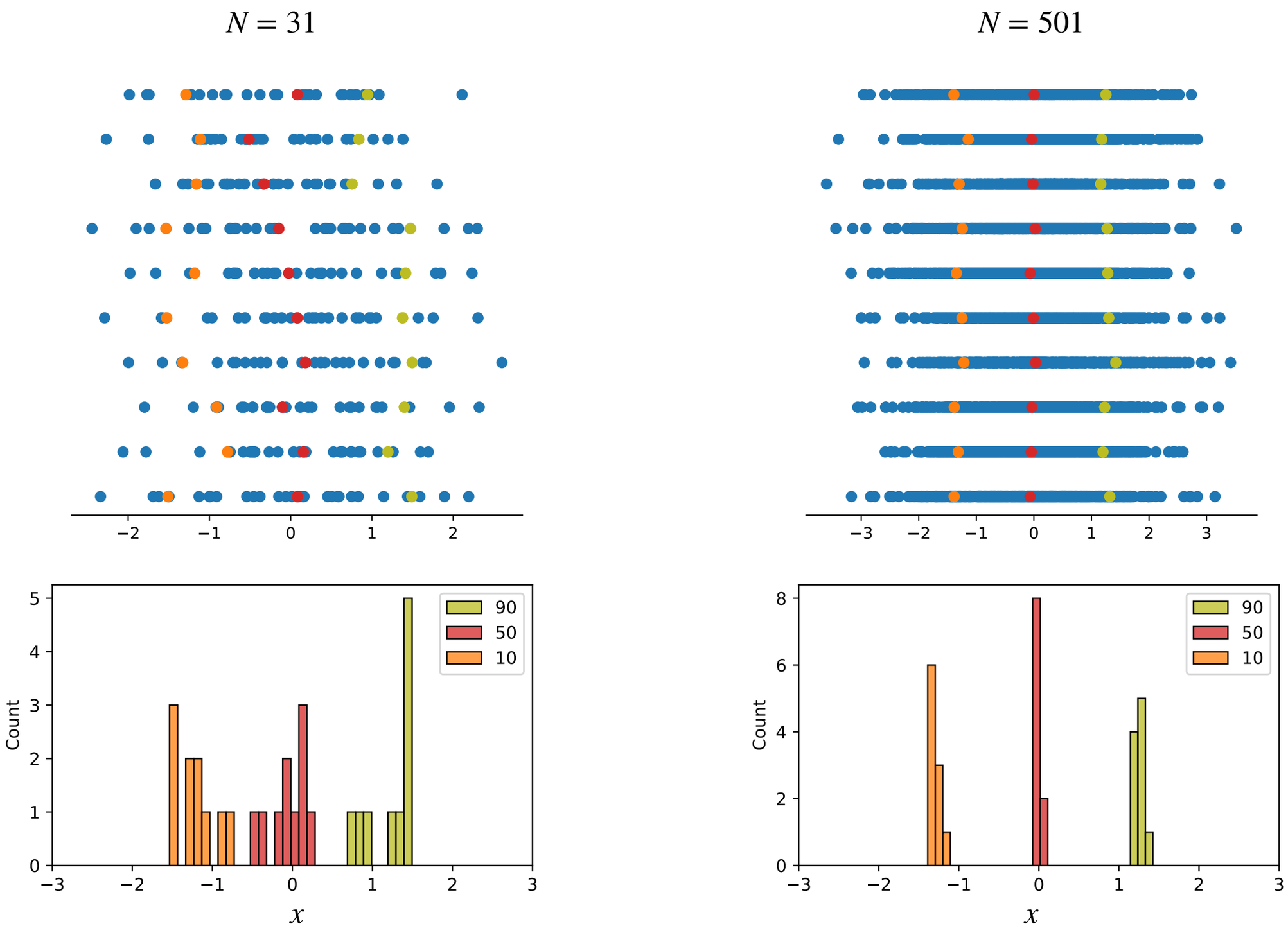
$$\boldsymbol{H}_n = \begin{pmatrix} \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{0} & \tilde{\boldsymbol{H}}_{\boldsymbol{n}} \end{pmatrix}$$

$$\boldsymbol{U} = \boldsymbol{H}_D(\boldsymbol{v}_D)\boldsymbol{H}_{D-1}(\boldsymbol{v}_{D-1})\ldots\boldsymbol{H}_1(\boldsymbol{v}_1)$$

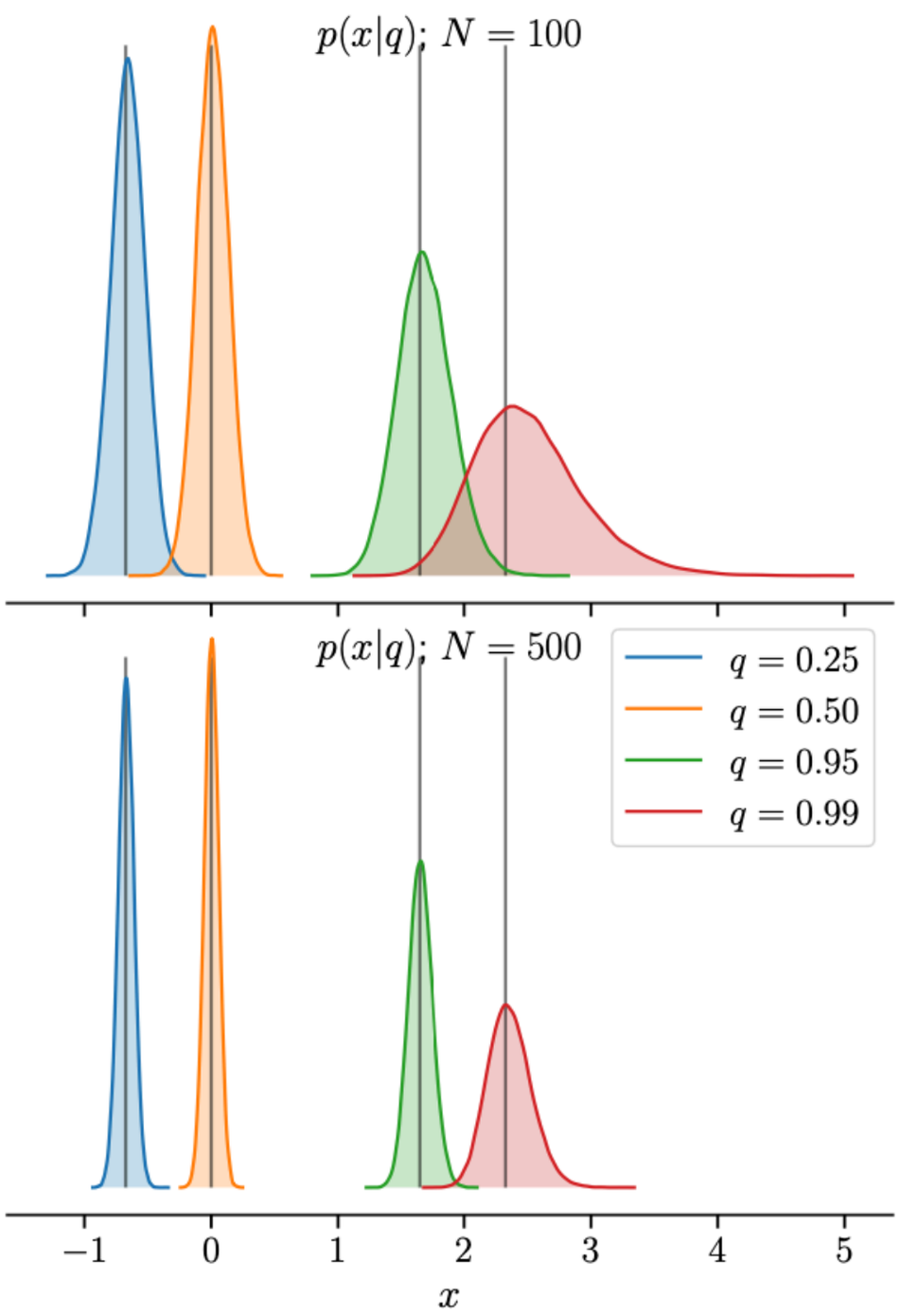


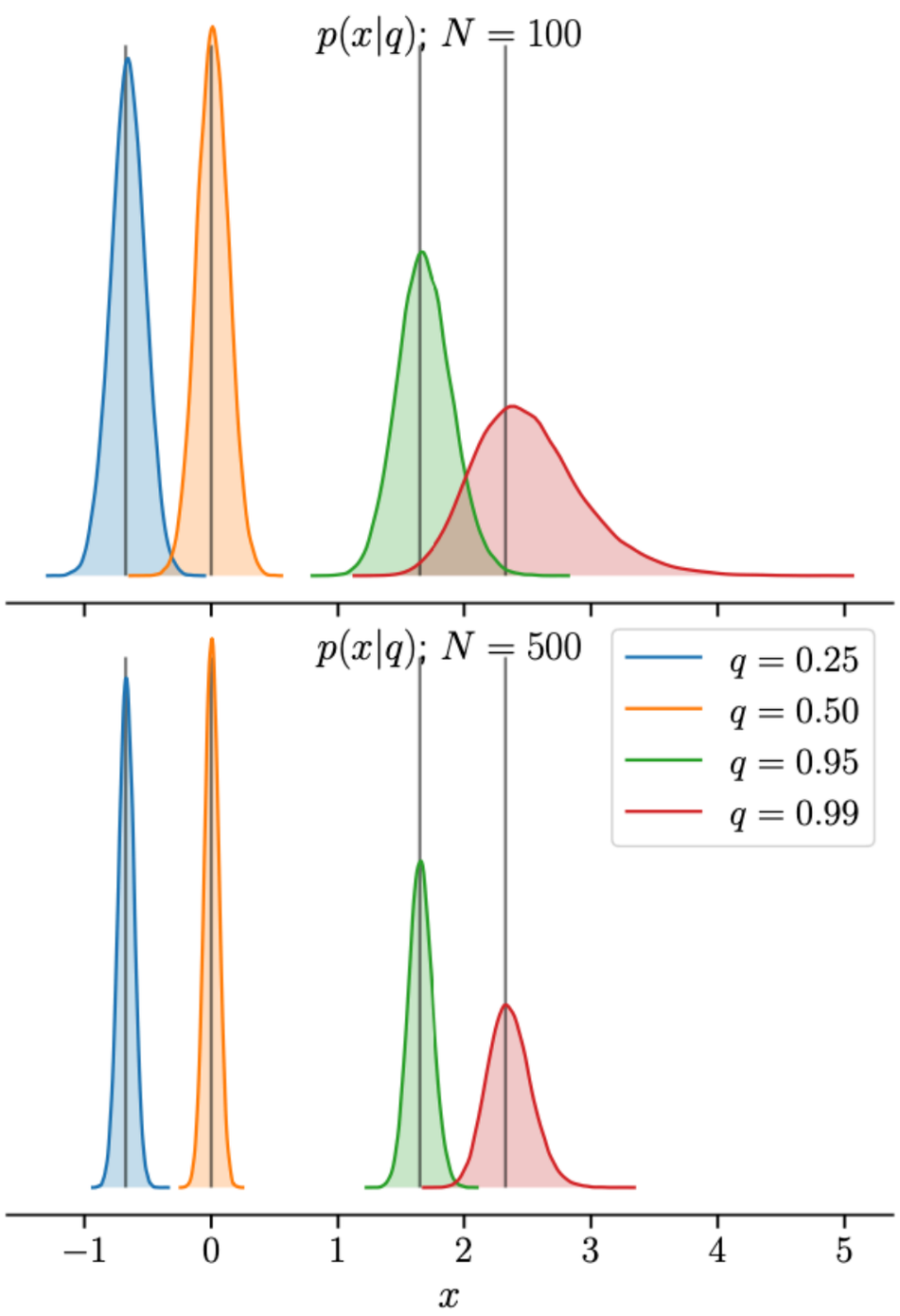
# Bayesian Quantile Matching Estimation



# Order Statistics



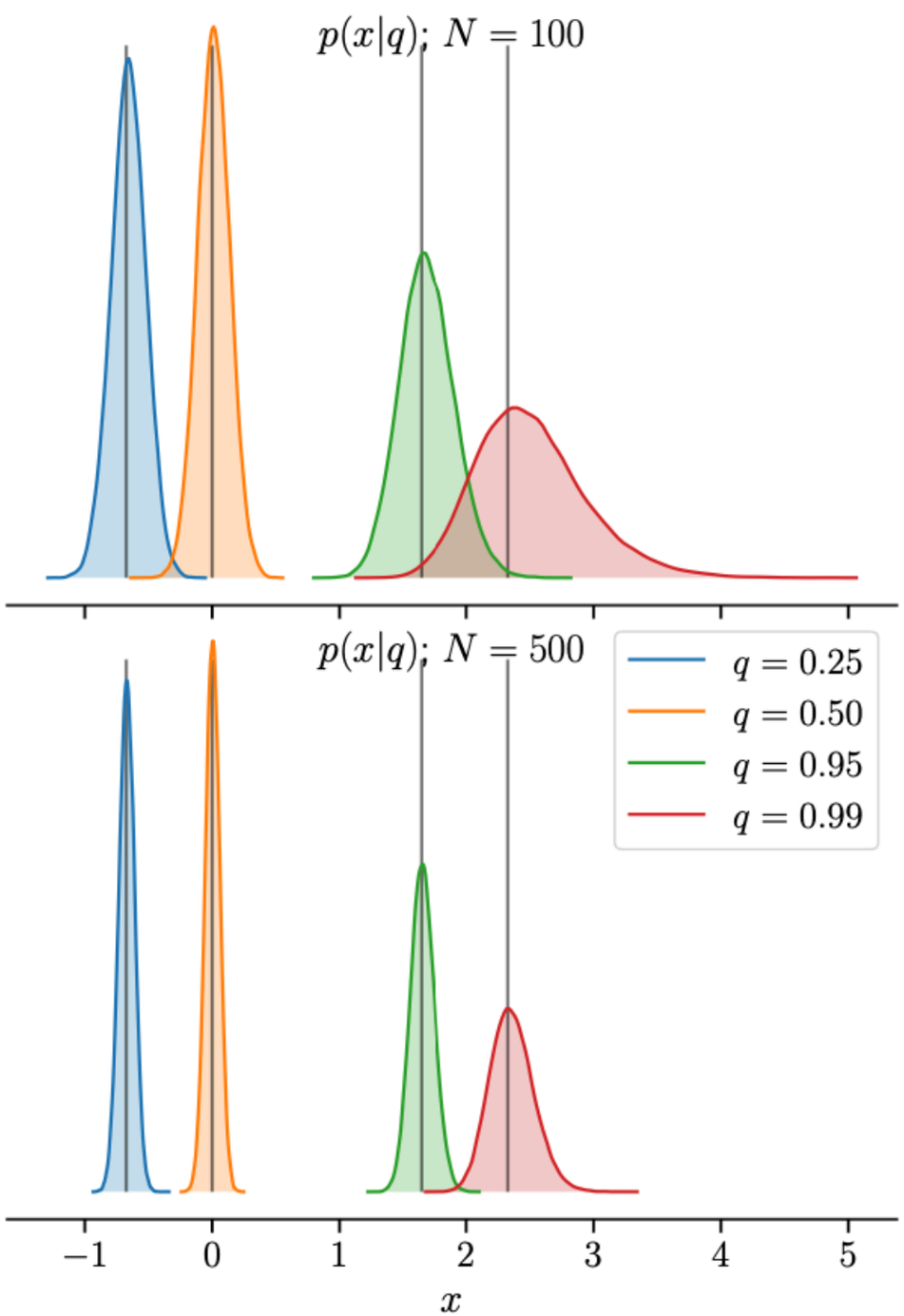
# Order Statistics




Order Statistics of Uniform distribution

$$p(X_k = x) = n\binom{n-1}{k-1}x^{k-1}(1-x)^{n-k}$$
$$= \mathrm{Beta}(x \,|\, k, n-k+1)$$

# Order Statistics




Order Statistics of Uniform distribution

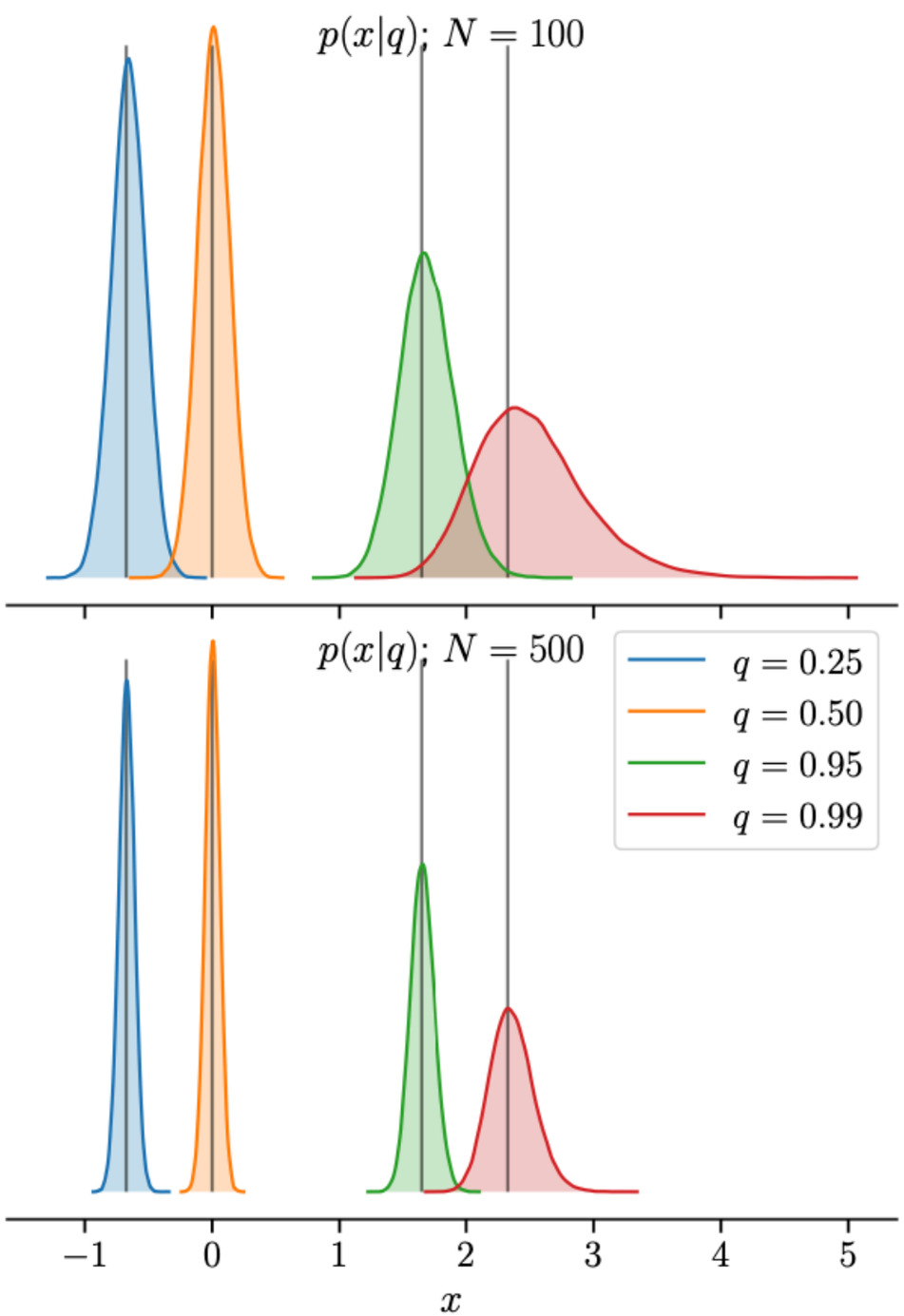
$$p(X_k = x) = n\binom{n-1}{k-1}x^{k-1}(1-x)^{n-k}$$
$$= \mathrm{Beta}(x\,|\,k, n-k+1)$$

Order Statistics of distribution with PDF $f$ and CDF $F$

$$p_x(X_k) = p_u\left(F_x(X_k)\right)\left|\frac{\mathrm{d}F_x(x)}{\mathrm{d}x}\right|_{x=X_k}$$

$$p_x(X_k) = \mathrm{Beta}\left(F_x(X_k)\,|\,k, n-k+1\right)f_x(X_k)$$

# Order Statistics




Order Statistics of Uniform distribution

$$p(X_k = x) = n\binom{n-1}{k-1}x^{k-1}(1-x)^{n-k}$$
$$= \mathrm{Beta}(x\,|\,k, n-k+1)$$

Order Statistics of distribution with PDF $f$ and CDF $F$

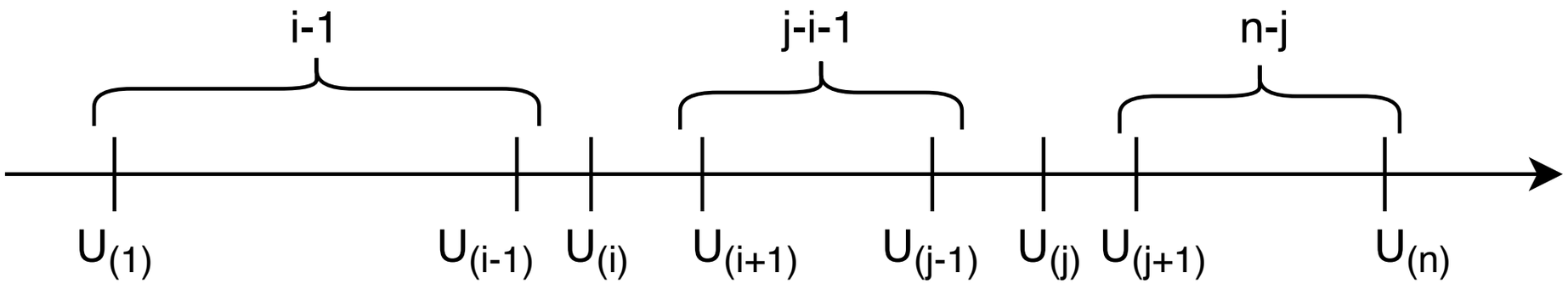
$$p_x(X_k) = p_u\left(F_x(X_k)\right)\left|\frac{\mathrm{d}F_x(x)}{\mathrm{d}x}\right|_{x=X_k}$$

$$p_x(X_k) = \mathrm{Beta}\left(F_x(X_k)\,|\,k, n-k+1\right)f_x(X_k)$$

Joint distribution of observed quantiles

$$p_X(x\,|\,q,\theta,N) = cF_\theta(x_1)^{q_1N-1}(1-F_\theta(x_M))^{N-q_MN}$$
$$\prod_{m=2}^{M}(F_\theta(x_m)-F_\theta(x_{m-1}))^{q_mN-q_{m-1}N-1}$$
$$\prod_{m=1}^{M}f_\theta(x_m)$$

# Joint distribution of the order statistics




Uniform

$$p_U(u\,|\,k) = \; c u_1^{k_1-1}(1-u_M)^{n-k_M}\prod_{m=2}^{M}(u_m-u_{m-1})^{k_m-k_{m-1}-1}$$

Non-Uniform (PDF $f$ and CDF $F$)

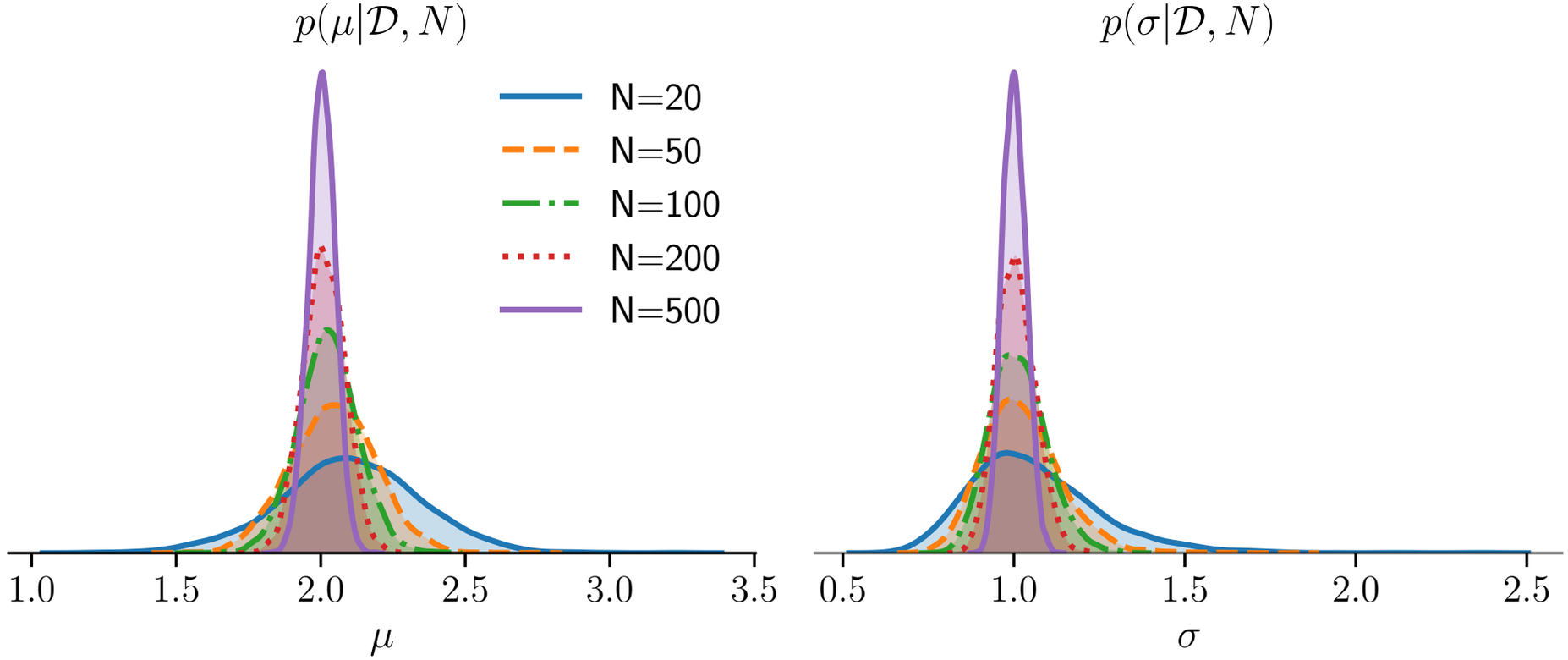
$$p_X(x\,|\,k) = cF(x_1)^{k_1-1}(1-F(x_M))^{n-k_M}\prod_{m=2}^{M}(F(x_m)-F(x_{m-1}))^{k_m-k_{m-1}-1}\prod_{m=1}^{M}f(x_m)$$

Using quantiles instead of order

$$p_X(x\,|\,q,\theta,N) = cF_\theta(x_1)^{q_1N-1}(1-F_\theta(x_M))^{N-q_MN}$$
$$\prod_{m=2}^{M}(F_\theta(x_m)-F_\theta(x_{m-1}))^{q_mN-q_{m-1}N-1}$$
$$\prod_{m=1}^{M}f_\theta(x_m)$$

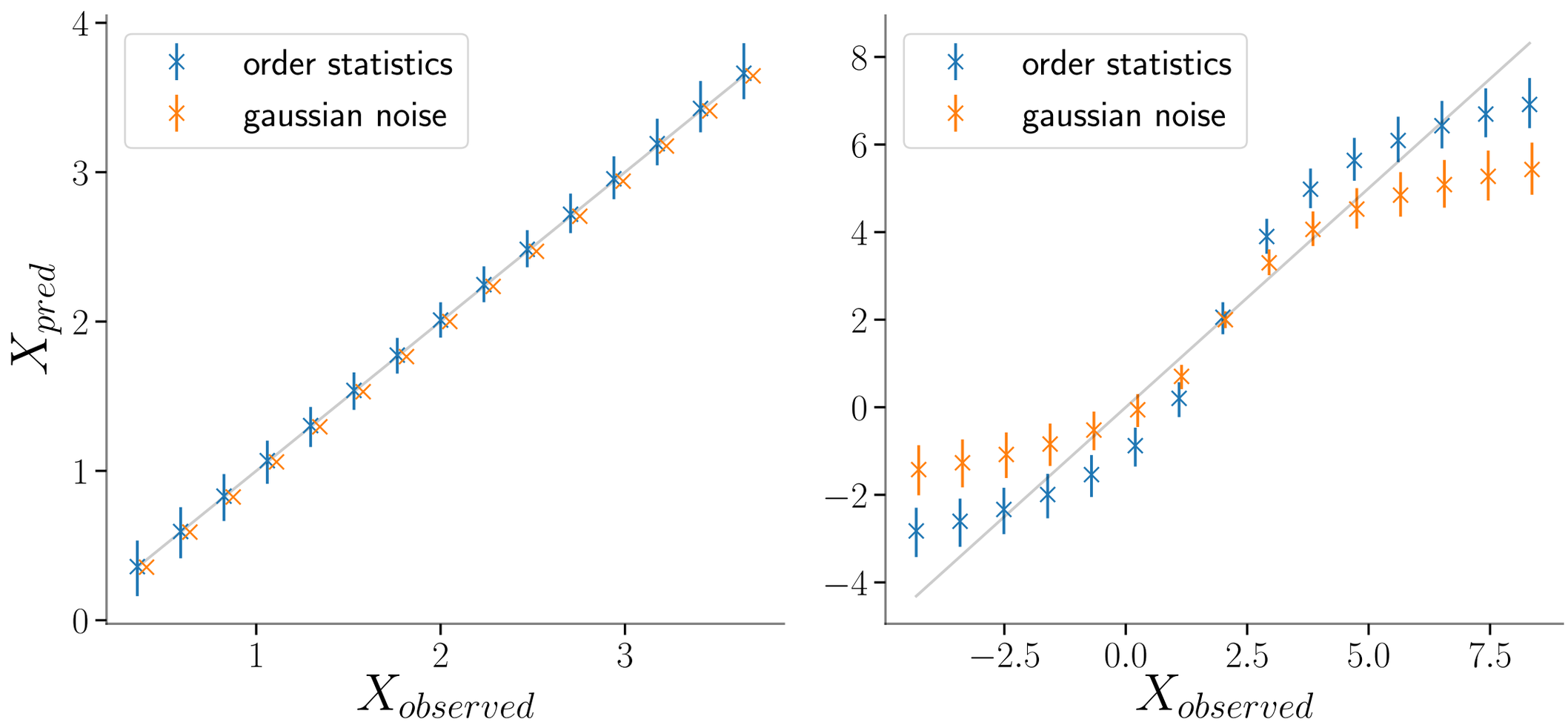
# Experiments

The higher $N$ the lower the uncertainty of the estimates



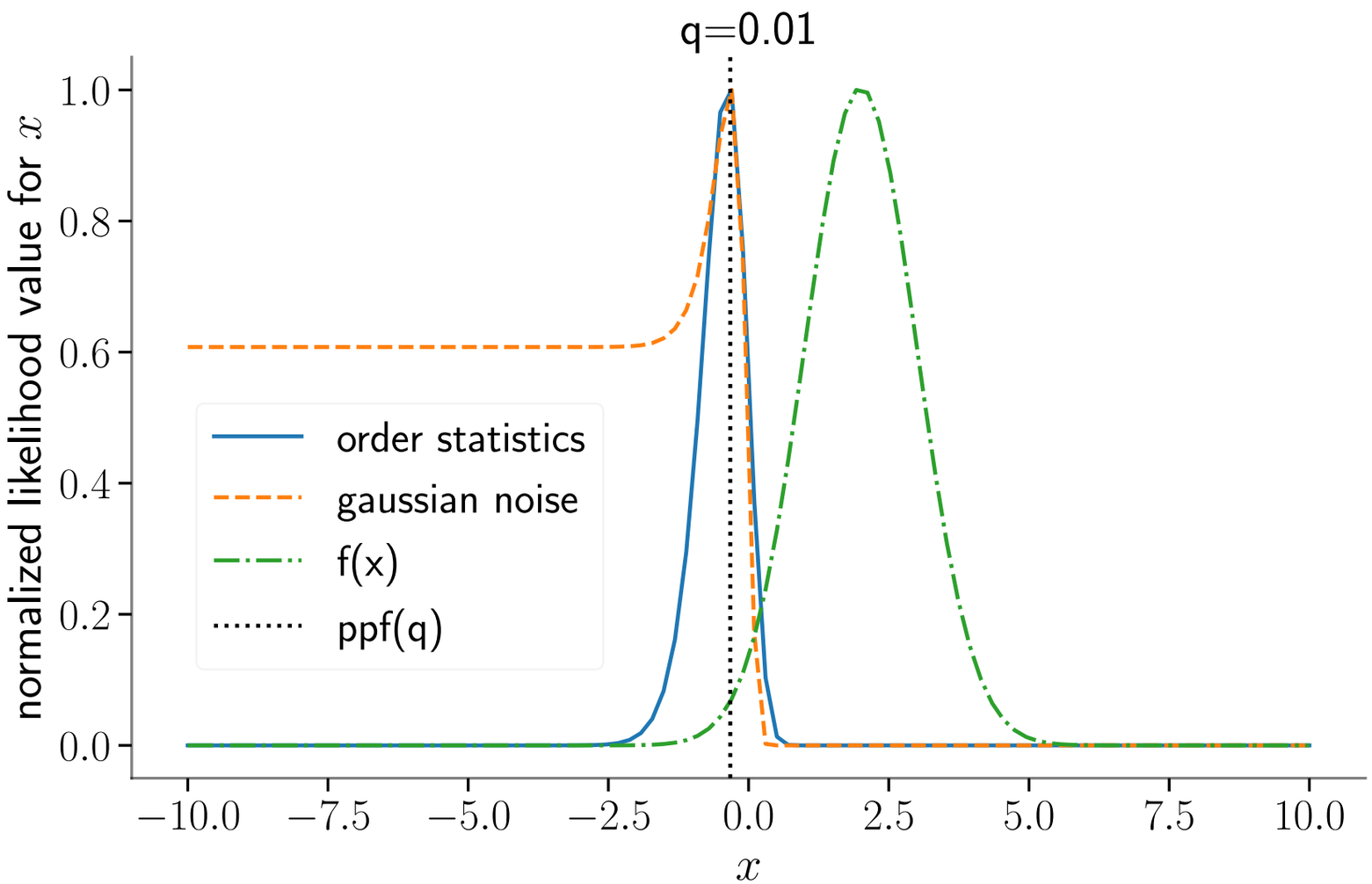
# Experiments

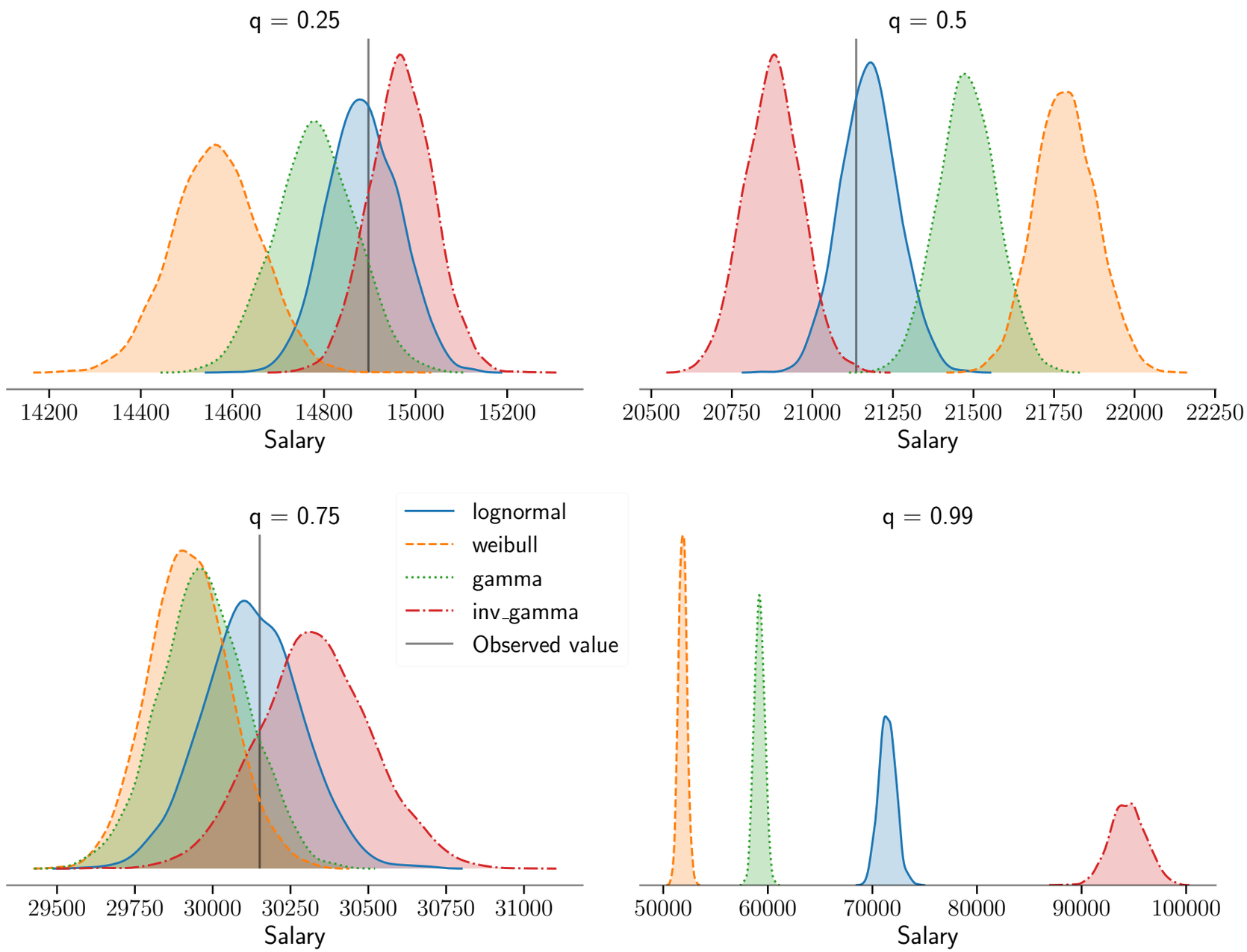
For unspecified model the tails are way better estimated by the order statistics than the gaussian noise model



# Experiments

Uncertainty estimation of a quantile in the tail



# Results

Fit for UK salary data using different models