Target tracking example

Filtering: $\left[X_t \middle| Y_{1:t}\right]$ (main interest)

Smoothing: $\left[X_{1:t} \middle| Y_{1:t}\right]$ (also given with SIS)

However as we have seen, the estimate of this distribution breaks down when t gets large due to the weights becoming degenerate (if we don't resample).

If we resample, most of the values sampled for X_1 will disappear when t gets large (related to the weight breakdown).

So SIS isn't useful for all problems.

Gibbs sampling

Special case of Markov Chain Monte Carlo (MCMC)

Instead of generating independent samples, it generates dependent samples via a Markov chain.

$$X^1 \rightarrow X^2 \rightarrow X^3 \rightarrow \ldots$$

Useful for a wide range of problems.

Popular for Bayesian analyses, but is a general sampling procedure. For example, it can be used to do smoothing in the target tracking example.

Similar to SIS in that the random variable X is decomposed into $X = \{X_1, X_2, \ldots, X_k\}$ and each piece is simulated separately.

However the conditioning structure is different. When sampling X_j, it is drawn conditional on all other components of X.

Gibbs sampler

A) Starting value: $X^0 = \{X_1^0, X_2^0, \ldots, X_k^0\}$

Picked by some mechanism

B) Sample $X^t = \{X_1^t, X_2^t, \ldots, X_k^t\}$ by

1) $X_1^t \sim \left[X_1 \middle| X_2^{t-1}, X_3^{t-1}, \ldots, X_k^{t-1} \right]$

2) $X_2^t \sim \left[X_2 \middle| X_1^{t}, X_3^{t-1}, \ldots, X_k^{t-1} \right]$

…

$$j)\quad X_j^t \sim \left[X_j \middle| X_1^t, \ldots, X_{j-1}^t, X_{j+1}^{t-1}, \ldots, X_k^{t-1} \right]$$

$\ldots$

$$k)\quad X_k^t \sim \left[X_j \middle| X_1^t, \ldots, X_{k-1}^t \right]$$

Under certain regularity conditions, the realizations $X^1, X^2, X^3, \ldots$ form a Markov chain with stationary distribution $[X]$.

Thus the realizations can be treated as dependent samples from the desired distribution.

Example: (Nuclear pump failure)

Gaver & O'Muircheartaigh (Technometrics, 1987)

Gelfand & Smith (JASA, 1990)

Observed 10 nuclear reactor pumps

Counted the number of failures for each pump

Pump	Failures (s_i)	Obs Time (t_i)	Obs Rate (l_i)
1	5	94.320	0.053
2	1	15.720	0.064
3	5	62.880	0.080
4	14	125.760	0.111
5	3	5.240	0.573
6	19	31.440	0.604
7	1	1.048	0.954
8	1	1.048	0.954
9	4	2.096	1.910
10	22	10.480	2.099

(Obs Time in 1000's of hours)

(Obs Rate = Failures / Time)

Want to determine the true failure rate for each pump with the following hierarchical model

$$s_i \mid \lambda_i \sim \text{Poisson}(\lambda_i t_i)$$
$$\lambda_i \mid \beta \sim \text{Gamma}(\alpha, \beta)$$
$$\beta \sim \text{IGamma}(\gamma, 1/\delta)$$

Note: $\beta \sim \text{IGamma}(\gamma, 1/\delta)$ is equivalent to

$$\frac{1}{\beta} \sim \text{Gamma}(\gamma, 1/\delta)$$

$$f(s_i \mid \lambda_i) = \frac{(\lambda_i t_i)^{s_i} \; e^{-\lambda_i t_i}}{s_i\,!}$$

$$\pi(\lambda_i \mid \beta) = \frac{\lambda_i^{\alpha-1} e^{-\lambda_i/\beta}}{\beta^{\alpha}\Gamma(\alpha)}$$

$$\rho(\beta) = \frac{\delta^{\gamma} e^{-\delta/\beta}}{\beta^{\gamma+1}\Gamma(\gamma)}$$

Want to determine

1) $[\lambda_i \mid S]$ for each pump i = 1, … , 10

2) $[\beta \mid S]$

where($S = \{s_1, \ldots, s_{10}\}$)

Note that both sets of these distributions are hard to get analytically.

Can show that

$$p(\lambda|\mathrm{S}) \propto \frac{1}{\left(\delta + \sum \lambda_i\right)^{10\alpha+\gamma}} \prod \frac{t_i^{\alpha+s_i} \lambda_i^{\alpha+s_i-1} e^{-\lambda_i t_i}}{\Gamma(\alpha + s_i)}$$

where $\lambda = \{\lambda_1, \ldots, \lambda_{10}\}$.

Note that the λ's are correlated and trying to get the marginal for each looks to be intractable analytically.

Run a Gibbs sampler to determine $[\lambda, \beta|\mathrm{S}]$. From this sampler we can get the desired distributions $[\lambda|\mathrm{S}]$ and $[\beta|\mathrm{S}]$.

A possible Gibbs scheme

Step 1) Sample $\lambda_1 \sim \left[\lambda_1 \middle| \lambda_{(-1)}, \beta, \mathrm{S}\right]$

…

Step 10) Sample $\lambda_{10} \sim \left[\lambda_{10} \middle| \lambda_{(-10)}, \beta, \mathrm{S}\right]$

Step 11) Sample $\beta \sim [\beta|\lambda, \mathrm{S}]$

where $\lambda_{(-j)} = \{\lambda_1, \ldots, \lambda_{j-1}, \lambda_{j+1}, \ldots, \lambda_{10}\}$

Need the following conditional distributions

$$\lambda_j \sim \left[\lambda_j \middle| \lambda_{(-j)}, \beta, S\right] = \left[\lambda_j \middle| \beta, s_j\right]$$

$$= \text{Gamma}\left(\alpha + s_j, \frac{1}{t_j + 1/\beta}\right)$$

$$\beta \sim \left[\beta \middle| \lambda, S\right] = \left[\beta \middle| \lambda\right]$$

$$= \text{IGamma}\left(\gamma + 10\alpha, \frac{1}{\delta + \sum \lambda_i}\right)$$

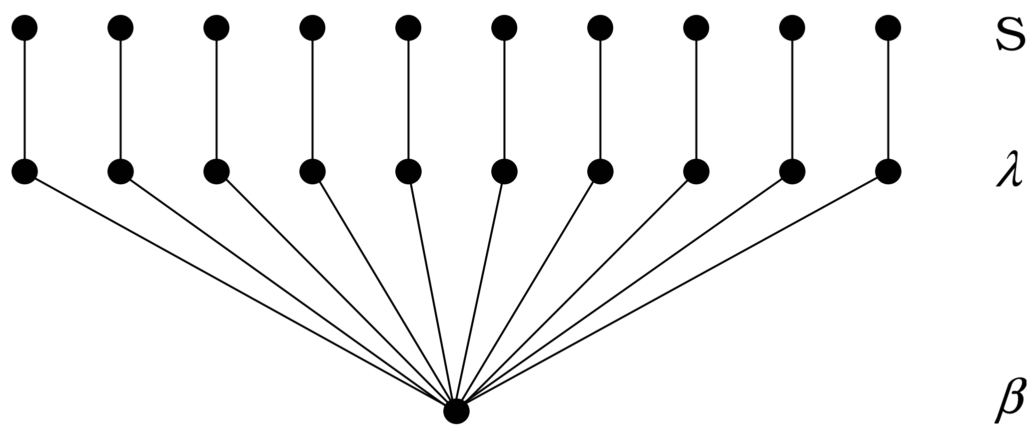
This can be gotten from the joint distribution by including only the terms in the product that contain the random variable of interest

$$[\lambda, \beta, S] = \left(\prod_{i=1}^{10} \frac{(\lambda_i t_i)^{s_i} e^{-\lambda_i t_i}}{s_i\,!}\right)\left(\prod_{i=1}^{10} \frac{\lambda_i^{\alpha-1} e^{-\lambda_i/\beta}}{\beta^{\alpha}\Gamma(\alpha)}\right)\frac{\delta^{\gamma} e^{-\delta/\beta}}{\beta^{\gamma+1}\Gamma(\gamma)}$$

e.g. for λ_j, which terms above have a λ_j in them.

Equivalently, you can do this by looking at the graph structure of the model by only including terms that correspond to edges joining to the node of interest.

e.g. for β, which edges connect with the node for β.



Example Run:

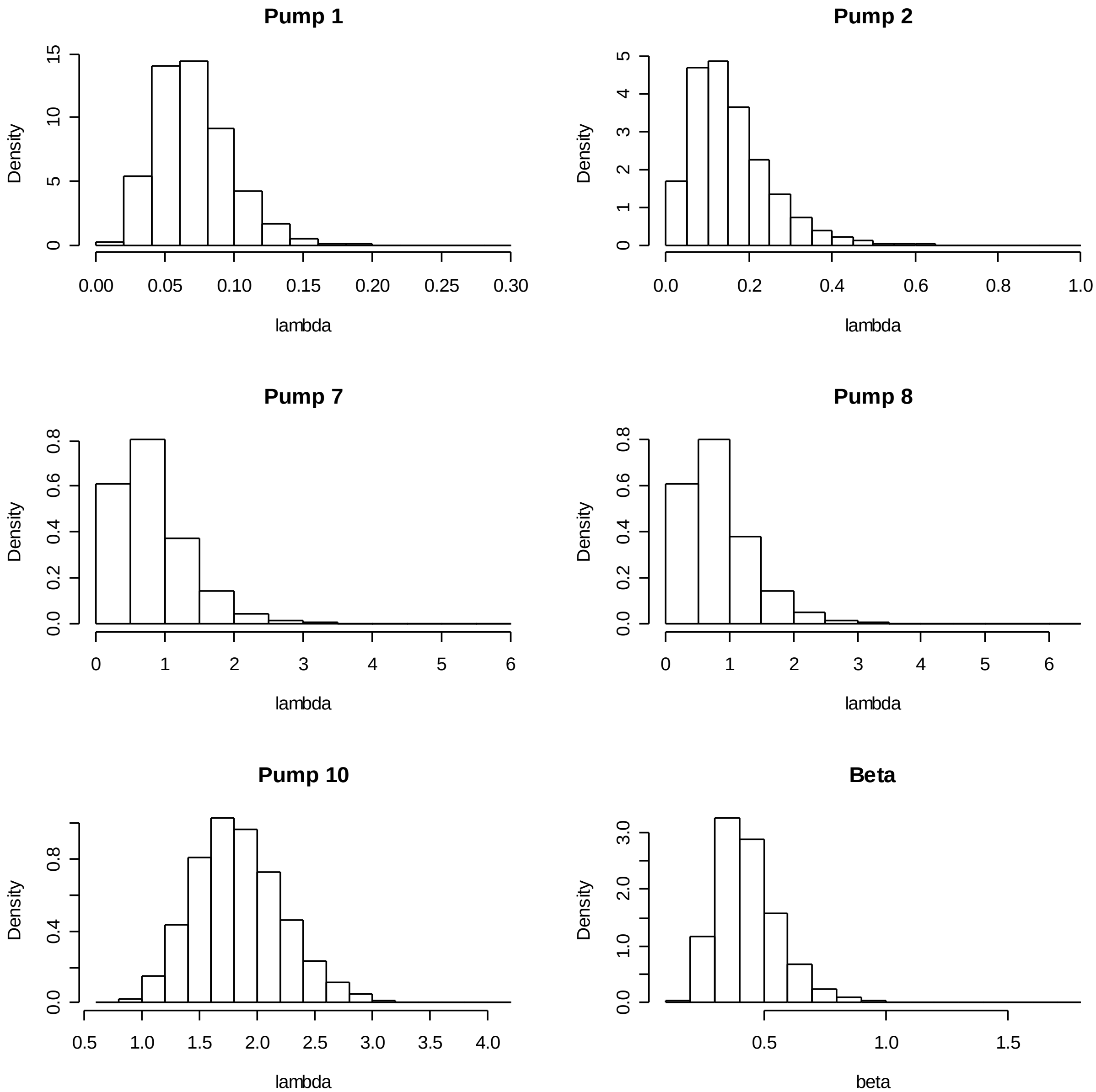
$\alpha = 1.8$

$\delta = 1$

$\gamma = 0.1$

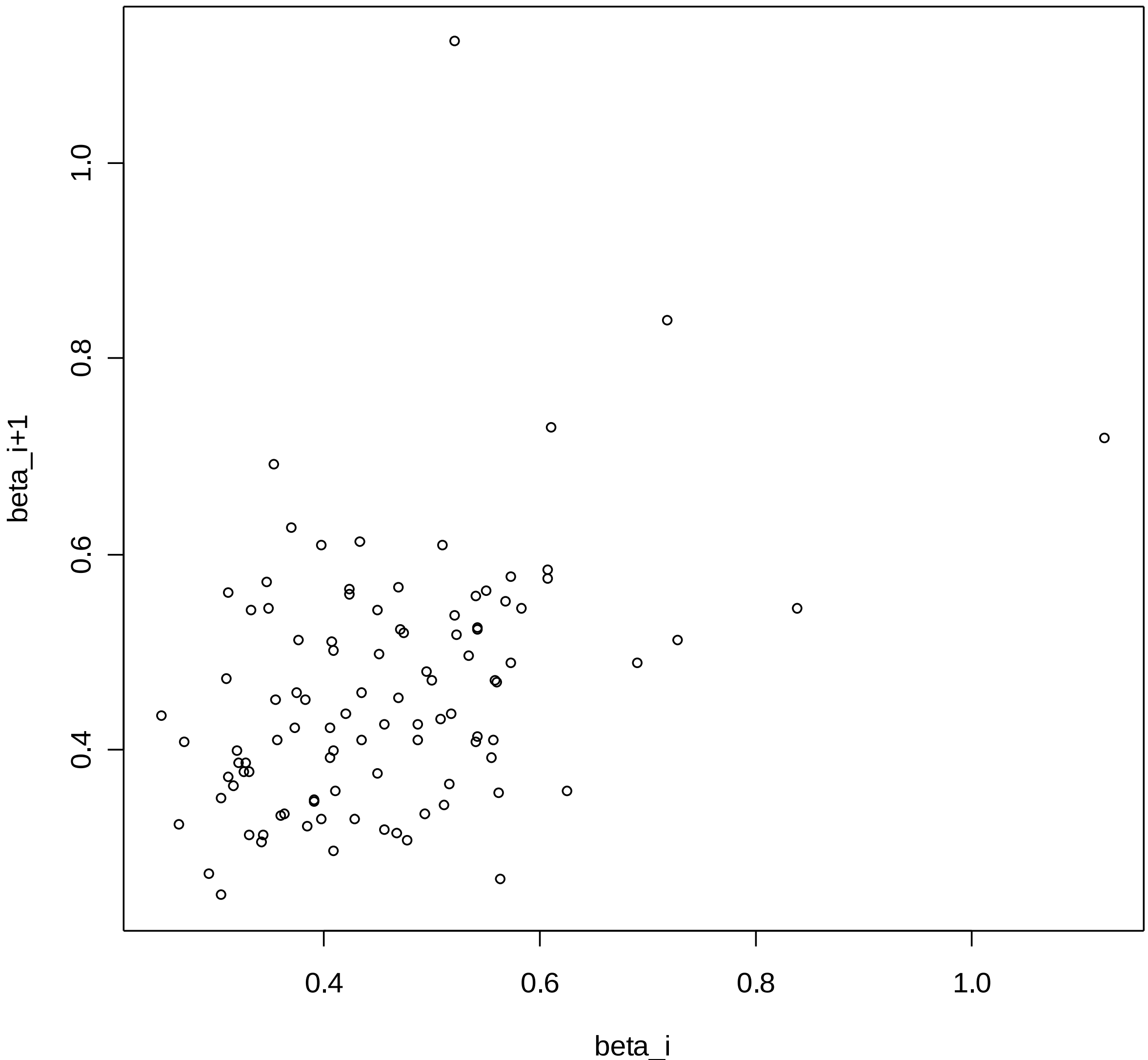
$n = 1000$

$\beta^0 = \bar{l}$

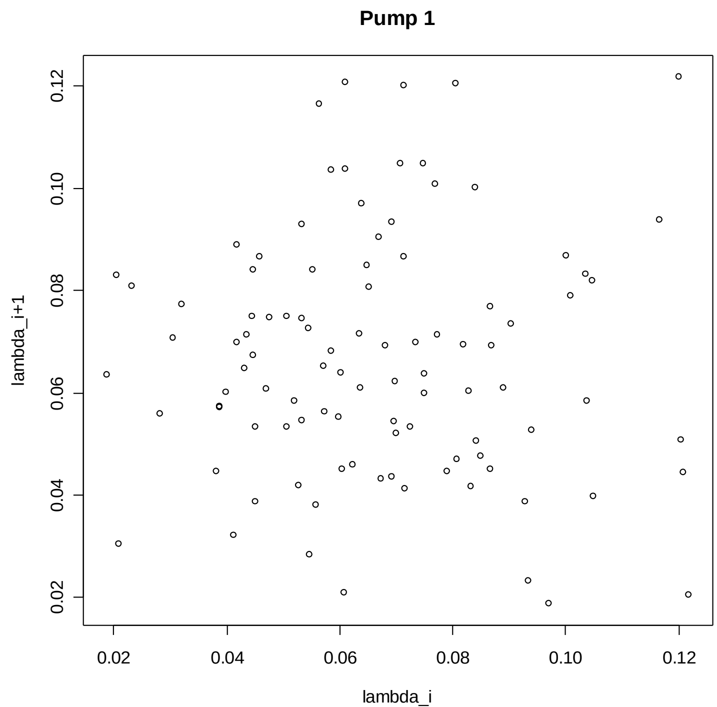

Pump 1
Density
lambda
0.00
0.05
0.10
0.15
0.20
0.25
0.30
0
5
10
15
Pump 2
Density
lambda
0.0
0.2
0.4
0.6
0.8
1.0
0
1
2
3
4
5
Pump 7
Density
lambda
0
1
2
3
4
5
6
0.0
0.2
0.4
0.6
0.8
Pump 8
Density
lambda
0
1
2
3
4
5
6
0.0
0.2
0.4
0.6
0.8
Pump 10
Density
lambda
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
0.0
0.4
0.8
Beta
Density
beta
0.5
1.0
1.5
0.0
1.0
2.0
3.0

Pump	Mean	Median	Std Dev
1	0.0702	0.0668	0.0268
2	0.1542	0.1363	0.0925
3	0.1039	0.0988	0.0399
4	0.1233	0.1206	0.0310
5	0.6263	0.5805	0.2924
6	0.6136	0.6040	0.1351
7	0.8241	0.7102	0.5267
8	0.8268	0.7129	0.5309
9	1.2949	1.2040	0.5776
10	1.8404	1.8121	0.3903

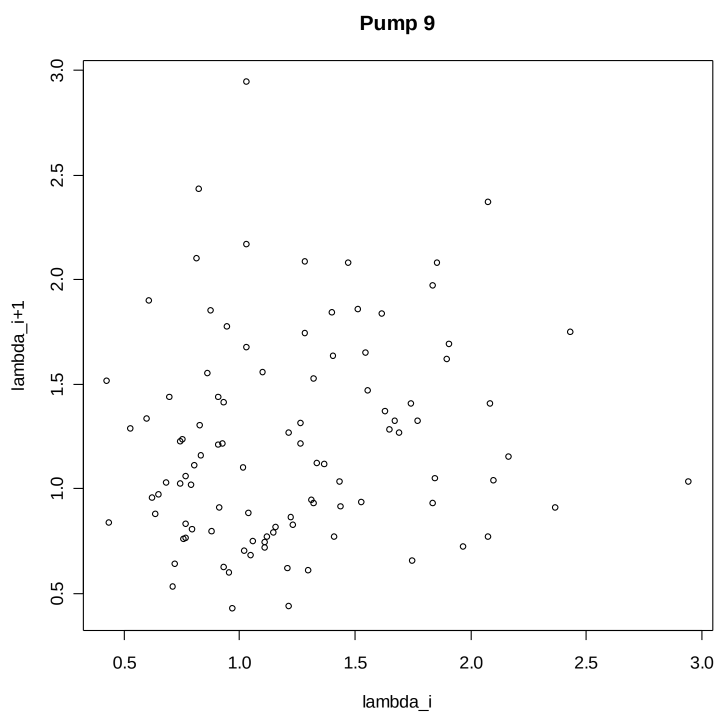
	Mean	Median	Std Dev
Beta	0.4372	0.4161	0.1315



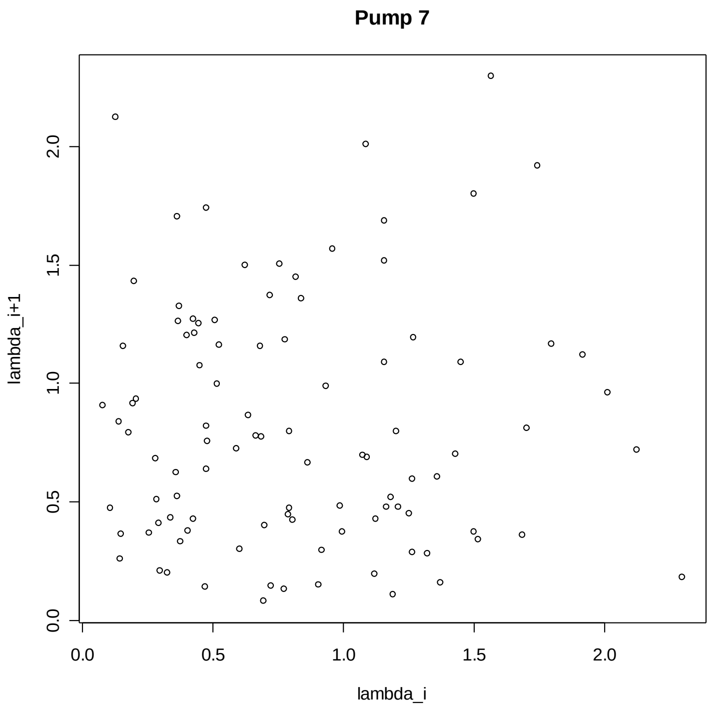
$$\mathrm{Cor}\left(\beta^{i}, \beta^{i+1}\right) = 0.302$$



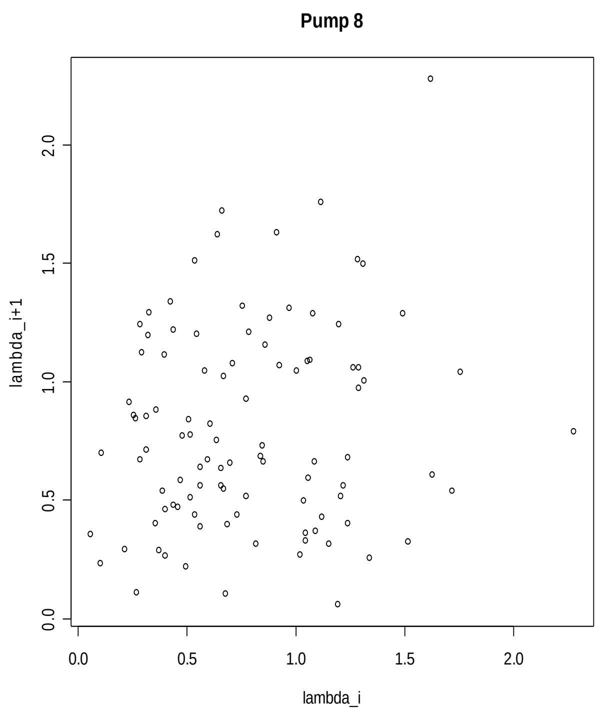
$$\mathrm{Cor}\left(\lambda_1^i, \lambda_1^{i+1}\right) = 0.012$$



$$\mathrm{Cor}\left(\lambda_9^i, \lambda_9^{i+1}\right) = 0.091$$



$$\mathrm{Cor}\left(\lambda_7^i, \lambda_7^{i+1}\right) = 0.063$$



$$\mathrm{Cor}\left(\lambda_8^i, \lambda_8^{i+1}\right) = 0.142$$