

KernelRobustness

Rasmus Brostrøm

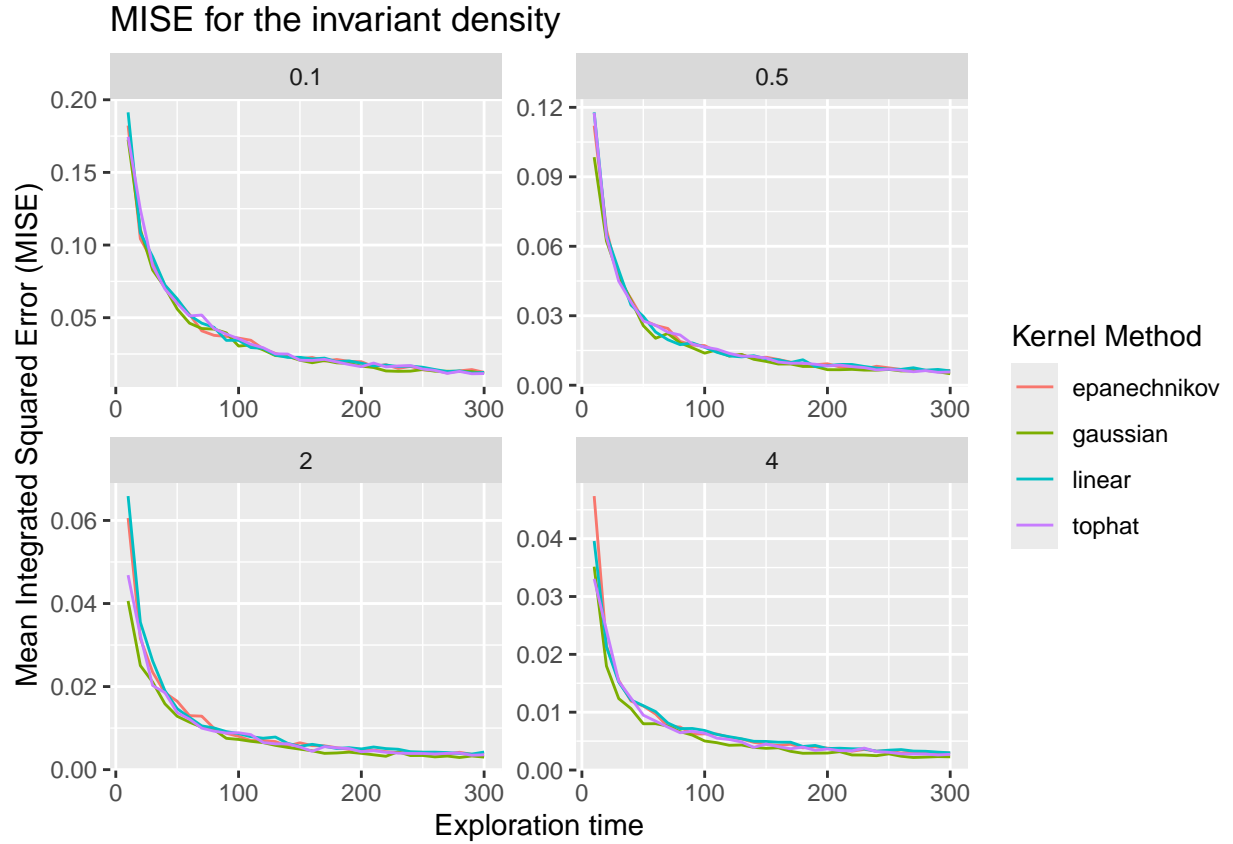
2024-04-16

MISE for pdf

First we can look at the mean integrated squared error (MISE) of the invariant density to see how changing the kernel affects the estimation of the true density. The reason, that we are looking at the MISE is that it gives a total overview of the risk associated with choice of kernel as it evaluates the deviations of the estimate to the true value over the whole support instead of evaluating it point wise. We are looking at four kernel methods: Epanechnikov, Gaussian, Linear (Uniform) and Tophat (Triangle) for four different drift functions:

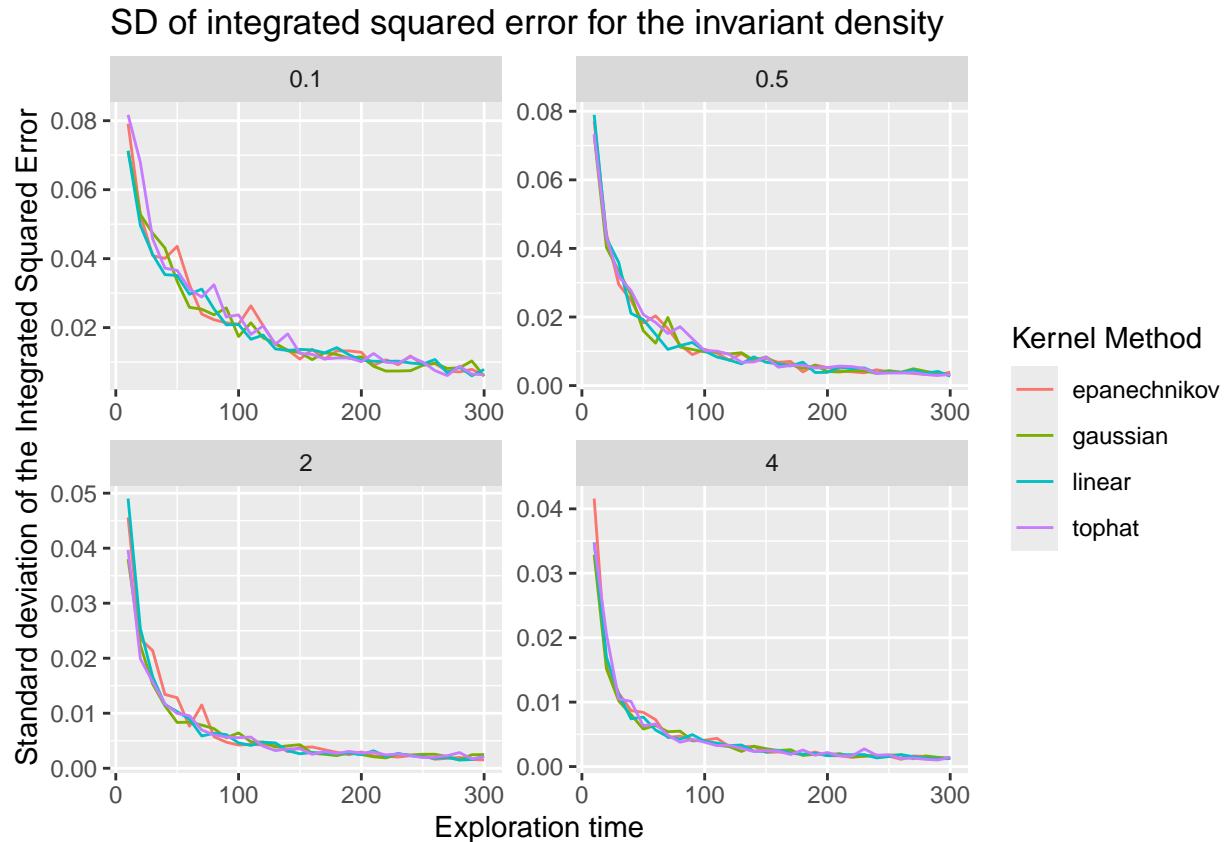
$$b(x) = -Cx, \quad C \in \{0.1, 0.5, 2, 4\}$$

and for exploration time S_T ranging from 10 to 300 with increments of 10. Each combination of kernel method, drift and exploration time is simulated 100 times and the average is calculated to approximate the MISE.



We can see that the choice of kernel method has little to no effect on the MISE for the different drift slopes. Each kernel method has almost the same trajectory when increasing the exploration time and it is only for the very short exploration time that the kernels deviate, where the Gaussian kernel seems to be the best choice of kernel function. Furthermore, we also see that the drift function affects the kernel methods ability to estimate the true invariant density, where a lower drift slope makes the estimated invariant density more inaccurate. The reason for this could have something to do with the fact that a higher drift slope makes the invariant density more concentrated around 0 (but why would this be better necessarily)?

Lets also see how the standard deviation of the integrated squared error changes with the choice of kernel function:

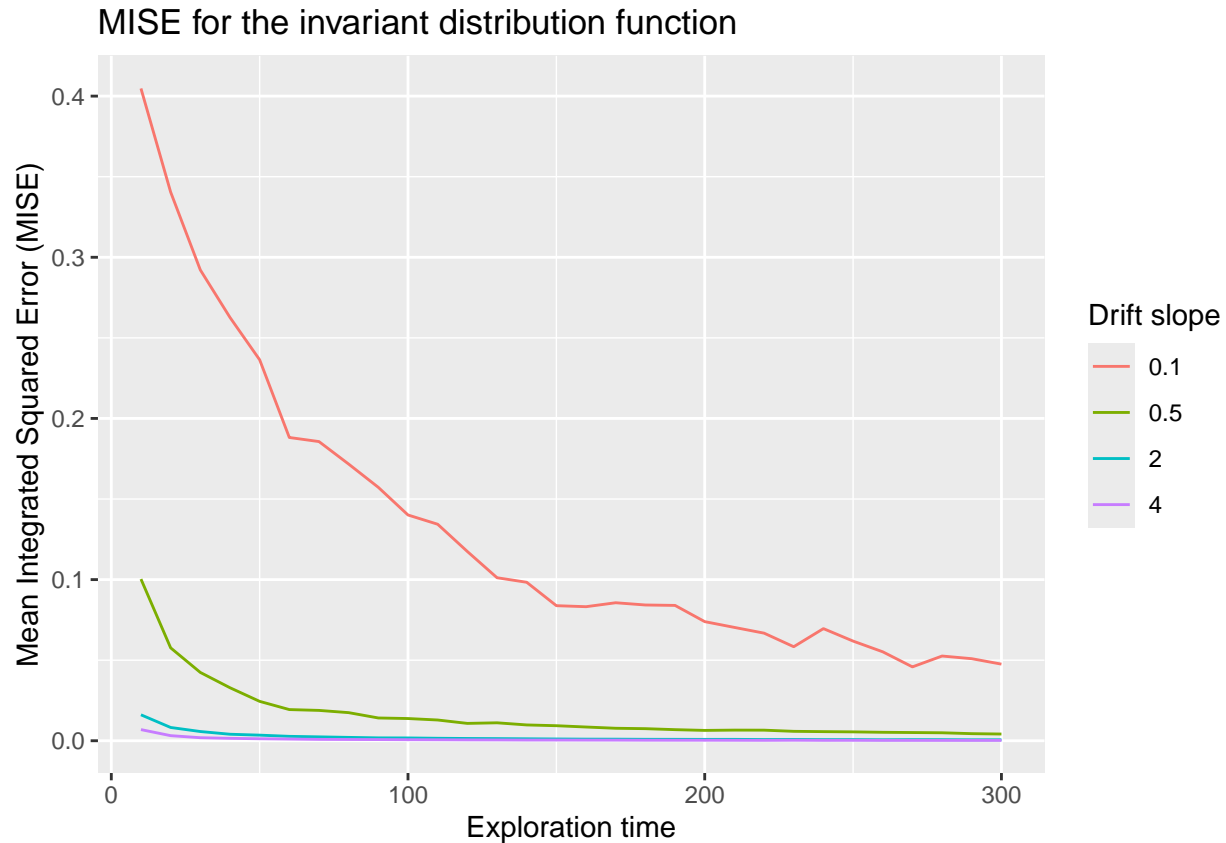


The plot shows that the all kernels basically have the same stability, when it comes to estimating the invariant density as they have the same trajectory for the different kernels. We also see again that the choice of drift function affects the stability of the kernel methods ability to estimate the invariant density, where a smaller drift slope increases the variability in the estimates of the invariant density. However, we also see that when the exploration time increases, then all kernels methods for all drift functions becomes more and more stable to the point where there is virtually no difference in the estimates for different sample paths of the diffusion process. Thus, if the kernel methods get enough data from the exploration periods, then they will all have close to the same precision, when estimating the invariant density.

We can therefore conclude that the choice of the kernel method for estimating the invariant density is irrelevant, and that any kernel method will have close to if not the same precision. This match the remark that is in the book about statistical inference for ergodic diffusion processes, where it is stated that the estimation of the invariant density using kernel density estimation is independent of the kernel method used, and that all kernel methods are (I believe) asymptotically consistent.

MISE for cdf

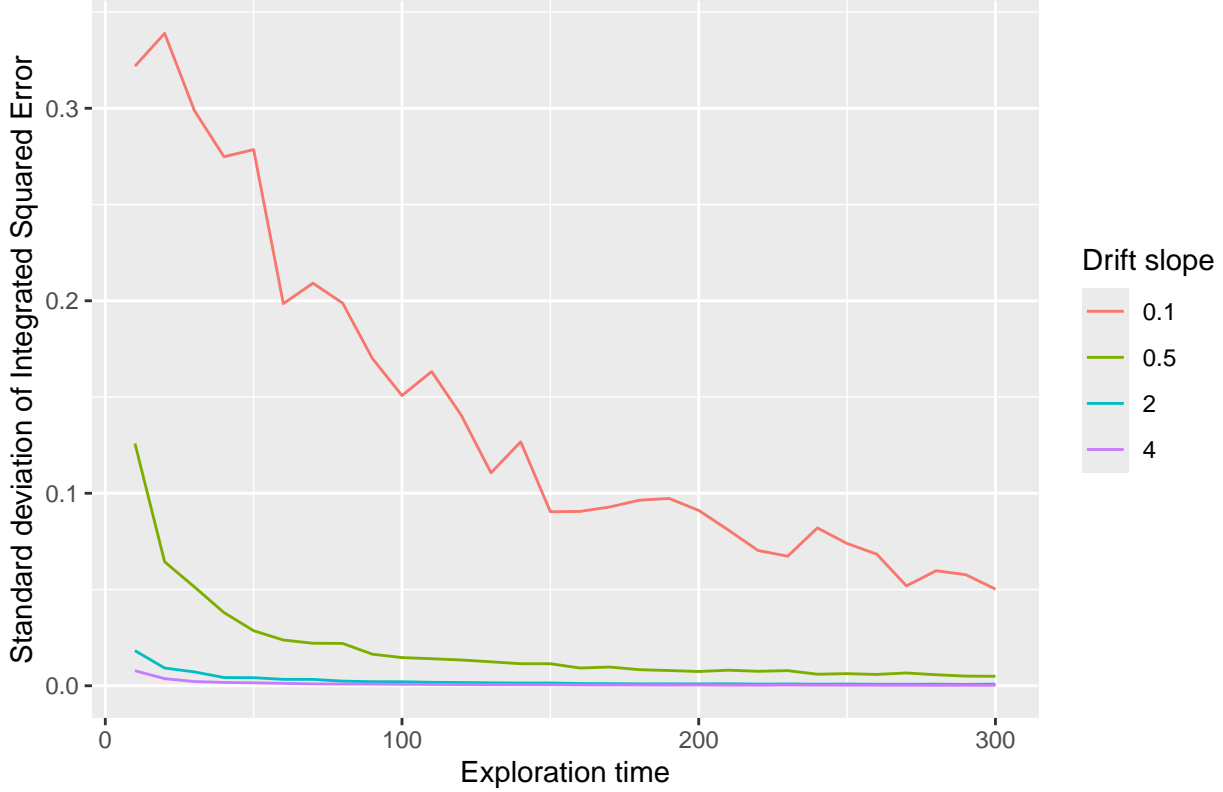
The invariant distribution function is estimated with the empirical distribution function, and does therefore not rely on any kernel method. Nonetheless, we will still look at how it performs with increasing exploration time and for the different drift functions.



As expected we see that the empirical distribution function gets closer to the true invariant distribution function as the exploration time increases, but we also see that the estimation of the invariant distribution is heavily affected by the drift slope. For the drift slopes of 2 and 4 there are no real difference in the estimate of the invariant distribution function, but as the drift slope decreases, then the empirical distribution function gets more inaccurate and needs more data to become accurate. Furthermore, even after an exploration time of 300 time units, then for the smallest drift slope of 0.1, the empirical distribution function is still not as accurate as it is for a drift slope of 0.5 and an exploration time of 50 time units.

Lets see how stable the empirical distribution function is for the different drift slopes.

SD for the integrated squared error for the invariant distribution function



Here we see that for a small drift slope the empirical distribution function is not only inaccurate for the different exploration times, but it is also more unstable. However, as with the mean of the integrated squared error, as the exploration time increases and the estimator gets more data, then the estimate becomes more stable, but still not to the point where it is as stable as for the other drift slopes.

We can therefore conclude that the estimation of the invariant distribution function for shorter exploration times are heavily relying on the drift slope of the drift function, and that a smaller drift slopes makes the estimation more inaccurate and more unstable.

Different kernels effect on strategy

Based the on the kernels ability to estimate the invariant density, then we wouldn't expect any difference in the performance of the algorithm, when changing kernel. Nonetheless, we still evaluate each kernel method presented above and their effect on the data-driven nonparametric algorithm. We do so by considering the drift functions:

$$b(x) = -Cx, \quad C \in \{0.5, 4\}$$

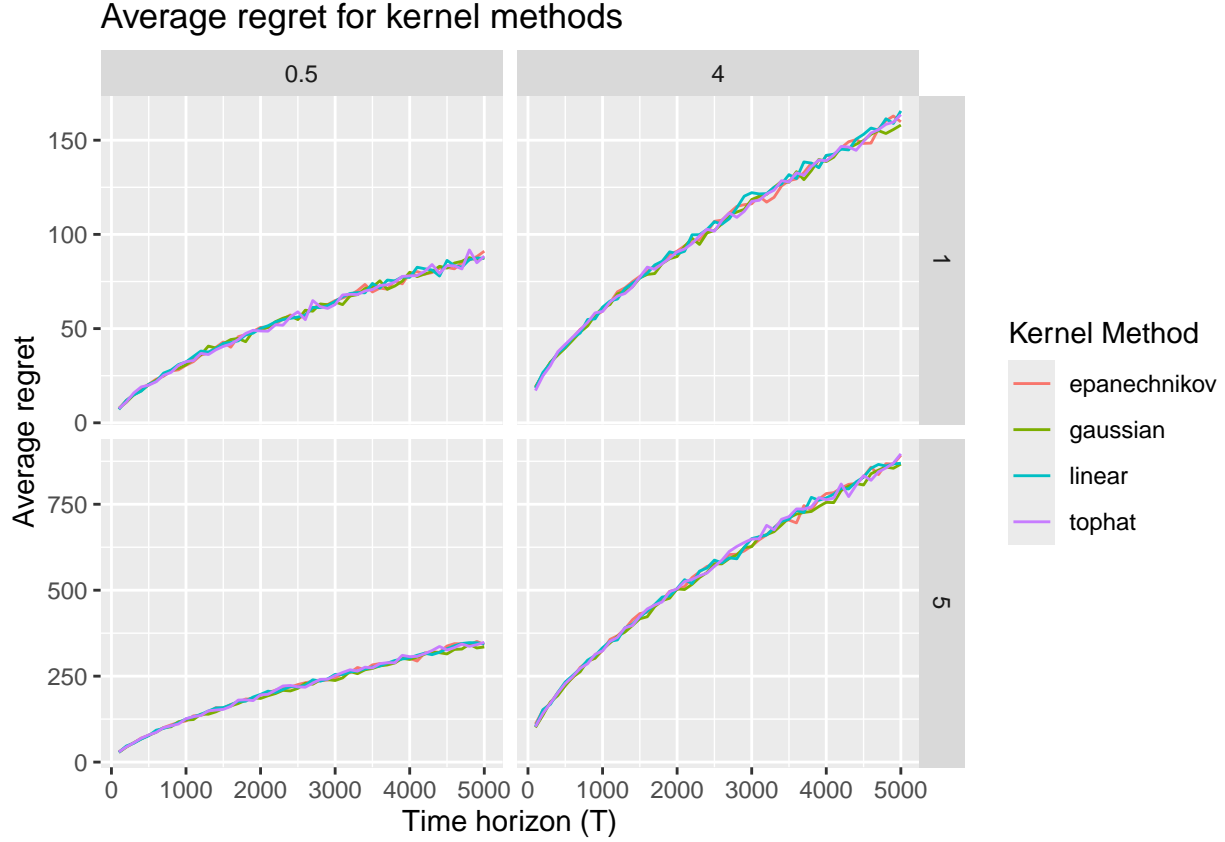
and the reward functions:

$$g(x) = 0.9 - |1 - x|^p \quad p \in \{1, 5\}$$

The four kernel methods are then run 100 times with time horizons $T \in \{100, 200, \dots, 5000\}$ for the four combinations of reward and drift function.

Average cumulative regret and variance of regret

Lets see how they perform in regards to the average cumulative regret:



We can clearly see that the only effect on the average regret comes from the change in drift and reward function, but that all kernel methods has the same average regret for each combination of reward and drift function. Lets also fit the average regret and see if the constants are the same. Here we also fit the regret to the form:

$$R = c \cdot T^p$$

Lets see the fits for the Gaussian kernel:

Reward power	c		p	
	Drift slope = 0.5	Drift slope = 4	Drift slope = 0.5	Drift slope = 4
1	0.394±0.0238	0.902±0.0402	0.636±0.0075	0.607±0.0055
5	1.34±0.0689	4.964±0.1773	0.651±0.0063	0.606±0.0044

For the Epanechnikov kernel:

Reward power	c		p	
	Drift slope = 0.5	Drift slope = 4	Drift slope = 0.5	Drift slope = 4
1	0.384±0.023	0.933±0.0461	0.64±0.0074	0.604±0.0061
5	1.374±0.0738	5.173±0.2372	0.651±0.0066	0.603±0.0057

For the tophat kernel:

Reward power	c		p	
	Drift slope = 0.5	Drift slope = 4	Drift slope = 0.5	Drift slope = 4
1	0.377 ± 0.0299	0.889 ± 0.0383	0.642 ± 0.0098	0.61 ± 0.0053
5	1.323 ± 0.0743	5.279 ± 0.2282	0.656 ± 0.0069	0.6 ± 0.0053

For the linear kernel:

Reward power	c		p	
	Drift slope = 0.5	Drift slope = 4	Drift slope = 0.5	Drift slope = 4
1	0.41 ± 0.0258	0.887 ± 0.0449	0.632 ± 0.0078	0.611 ± 0.0063
5	1.387 ± 0.0554	5.207 ± 0.2367	0.65 ± 0.0049	0.602 ± 0.0056

We can see from these tables that there are basically no difference in the fits and therefore the kernel method has no effect on the average cumulative regret.

Lets see if all kernel methods are just as stable:



Here we also see that the standard deviation of the regret is almost the same for each of the kernel methods for the four combinations of the reward and drift functions, with the Gaussian kernel having a slightly lower standard deviation for most of the time horizons when the power is high for the reward function. However, this is such a small difference and it is not consistent for all time horizons, thus making it irrelevant which kernel method is chosen.

For good measure, then lets also just quickly see if there is any change in the difference in number of decisions made made by the optimal strategy and the data-driven strategy, when changing kernel:



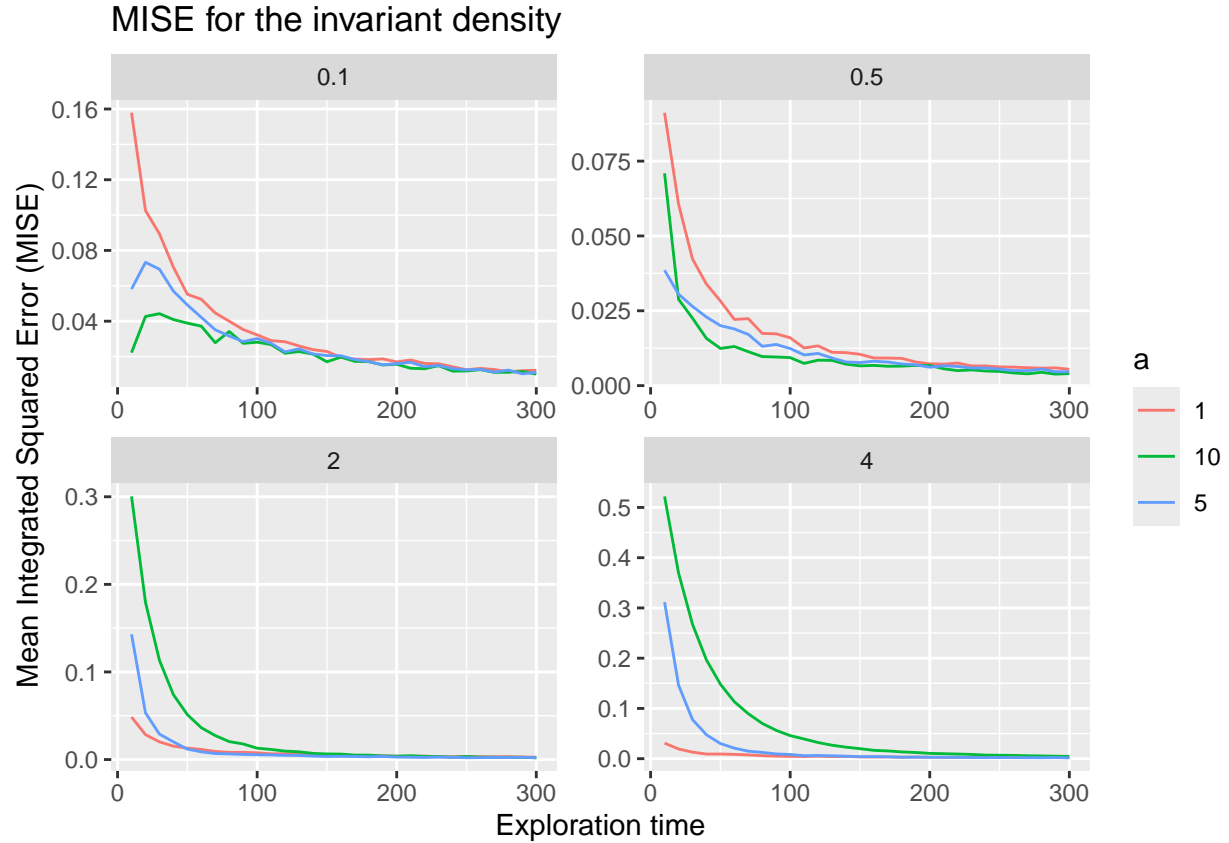
As expected we see that there is no difference in the average difference in number of decisions made by the optimal strategy and the data-driven strategy.

Different bandwidths

Different constants of the bandwidth

In this section we explore how the bandwidth affects the kernel density estimation of the invariant density for the gaussian kernel. To do so we will try different constants for the bandwidth of the form:

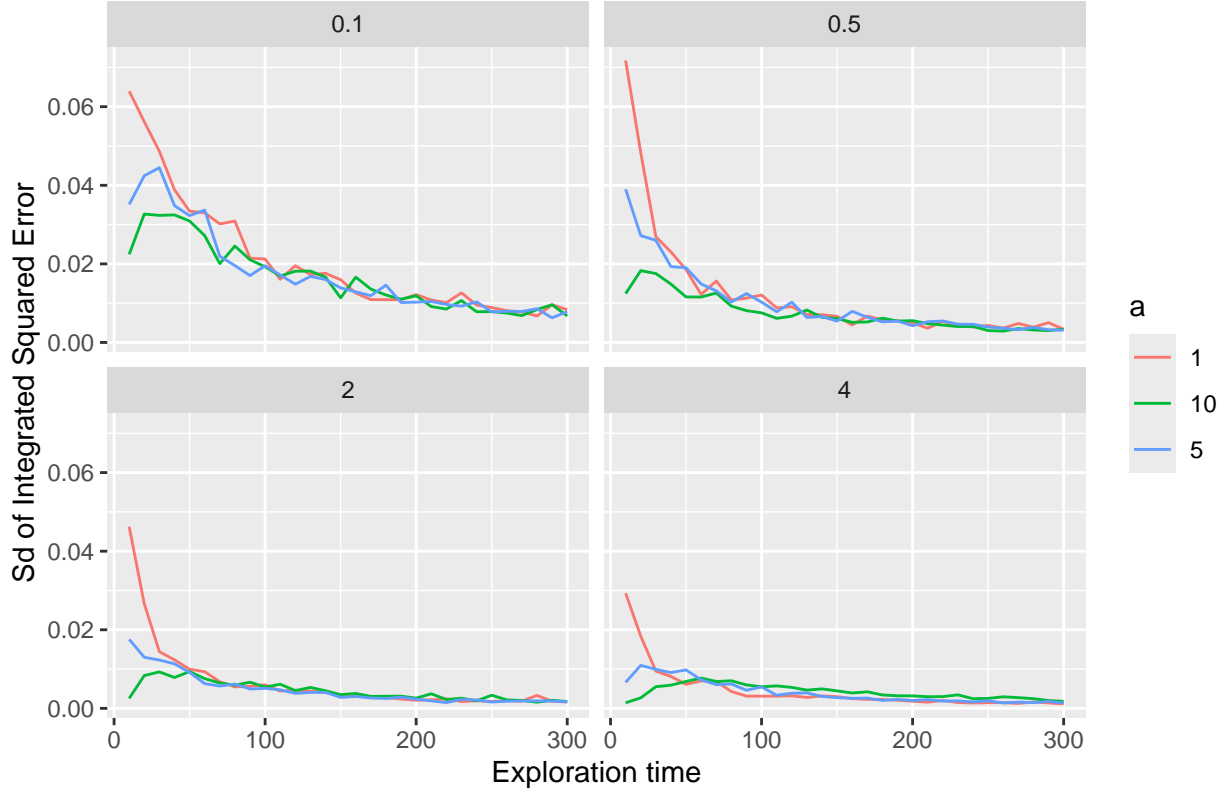
$$bw = \frac{a}{\sqrt{T}}, \quad a \in \{1, 5, 10\}$$



We can see that for low values of the drift slope coefficient, then the higher constant in the bandwidth makes the fit better for shorter exploration times on average. However, as the exploration time increases to about 100 time units, then there isn't really any difference in the fit for the different bandwidths. The increase in the constant of the bandwidth does however have a significant downside when the drift slope coefficient gets higher. Here we see that the increase in the constant of the bandwidth makes the fit worse for shorter exploration times and as the drift slope increases it takes longer time for a higher bandwidth constant to be able to have the same fit.

Lets see how stable the fit is for the different constants.

Standard deviation of Integrated Squared Error for the invariant density



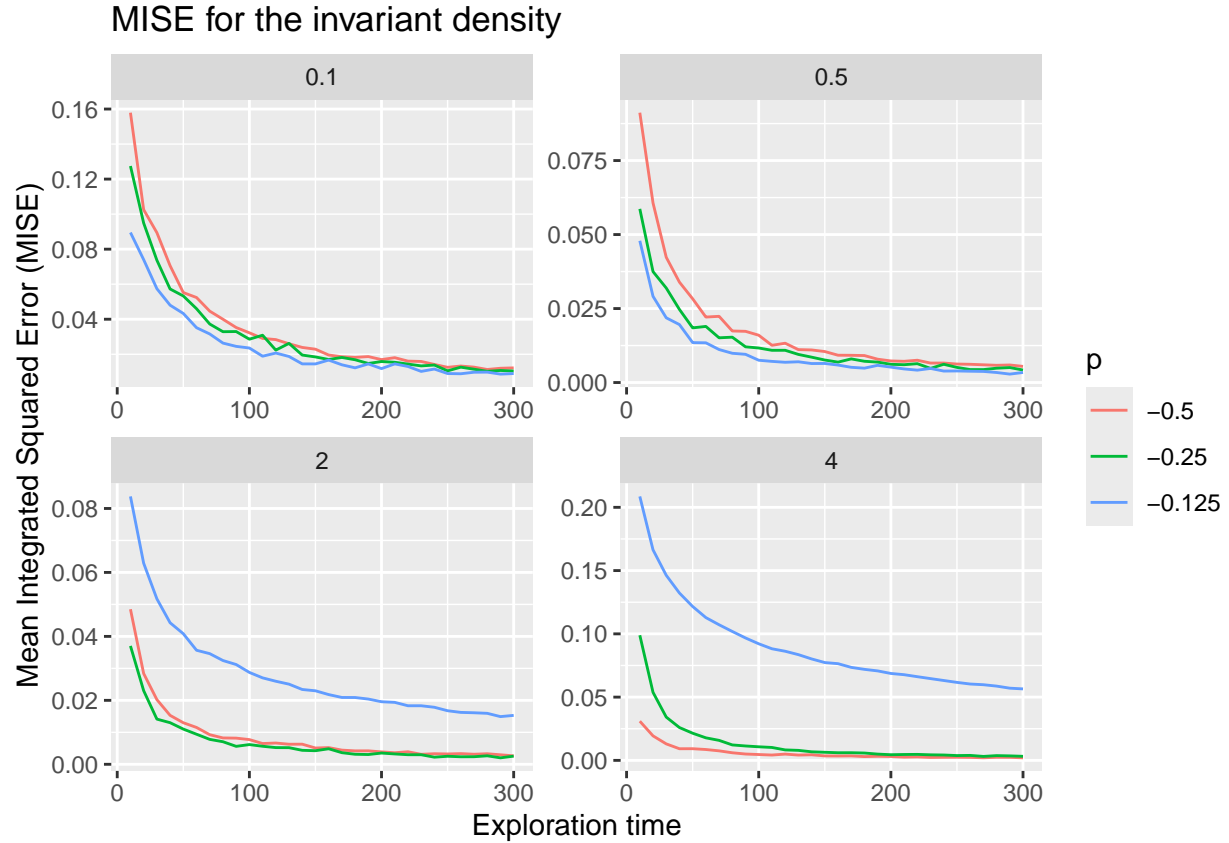
Here we can see that for all drift slope coefficients the higher bandwidth constant makes the integrated squared error and therefore the fit more stable when the exploration time is low, but also that an exploration time of about 50 time units, then all of the bandwidths have the same stability of the integrated squared error and therefore the same stability of their fits to the true invariant density.

Different powers of the bandwidth

Next we will see how changing the power of the bandwidth affects the Gaussian kernels ability to estimate the invariant density.

Here we are looking at the bandwidths of the form:

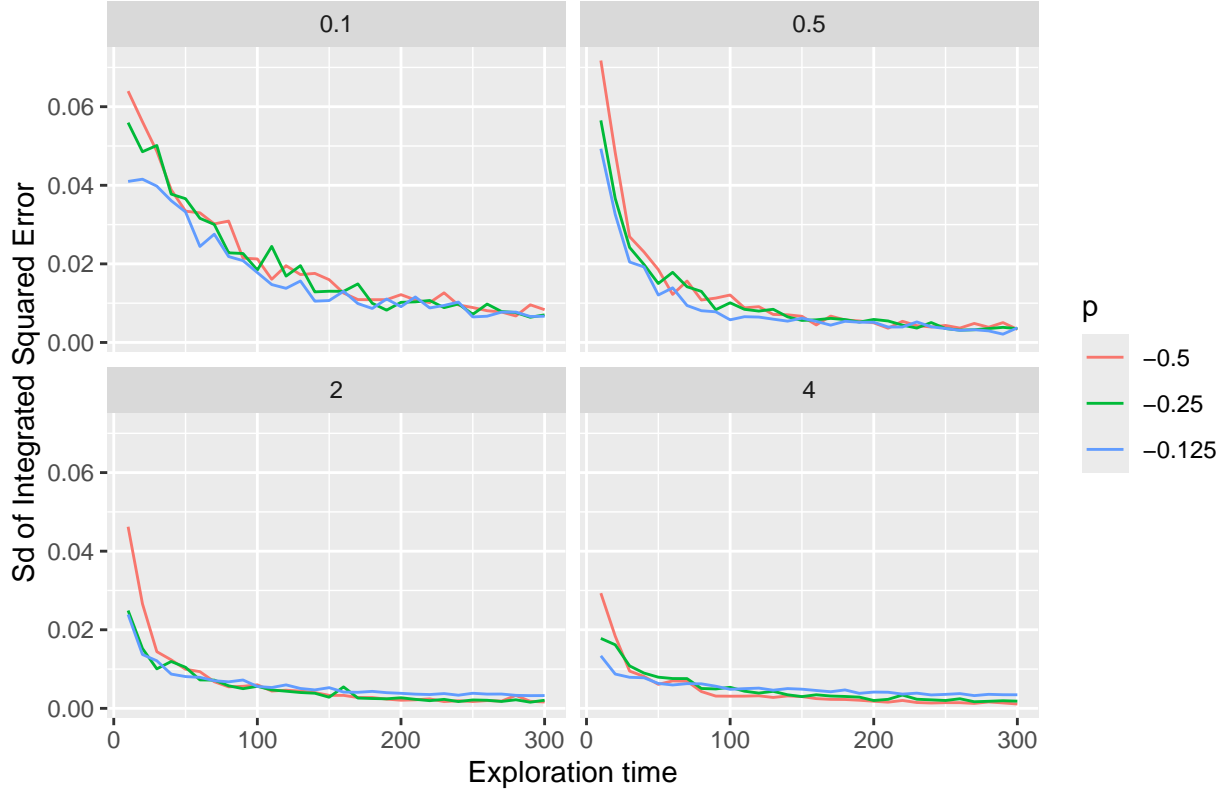
$$bw = T^p, \quad p \in \{-1/2, -1/4, -1/8\}$$



Here we see again see that for low drift slope coefficients, the higher powers of the bandwidth are slightly better, but that the difference is so small that it wouldn't make much difference. However, when the drift slope coefficient increases, then the higher power of the bandwidth makes the fit worse to the point where within the 300 time units of exploring, the power of -0.125 never achieves a fit as good as the power of -0.5 with only 10 time units of exploring for a drift slope coefficient of 4.

Lets look at the stability of estimates for the invariant density through the standard deviation of the integrated squared error:

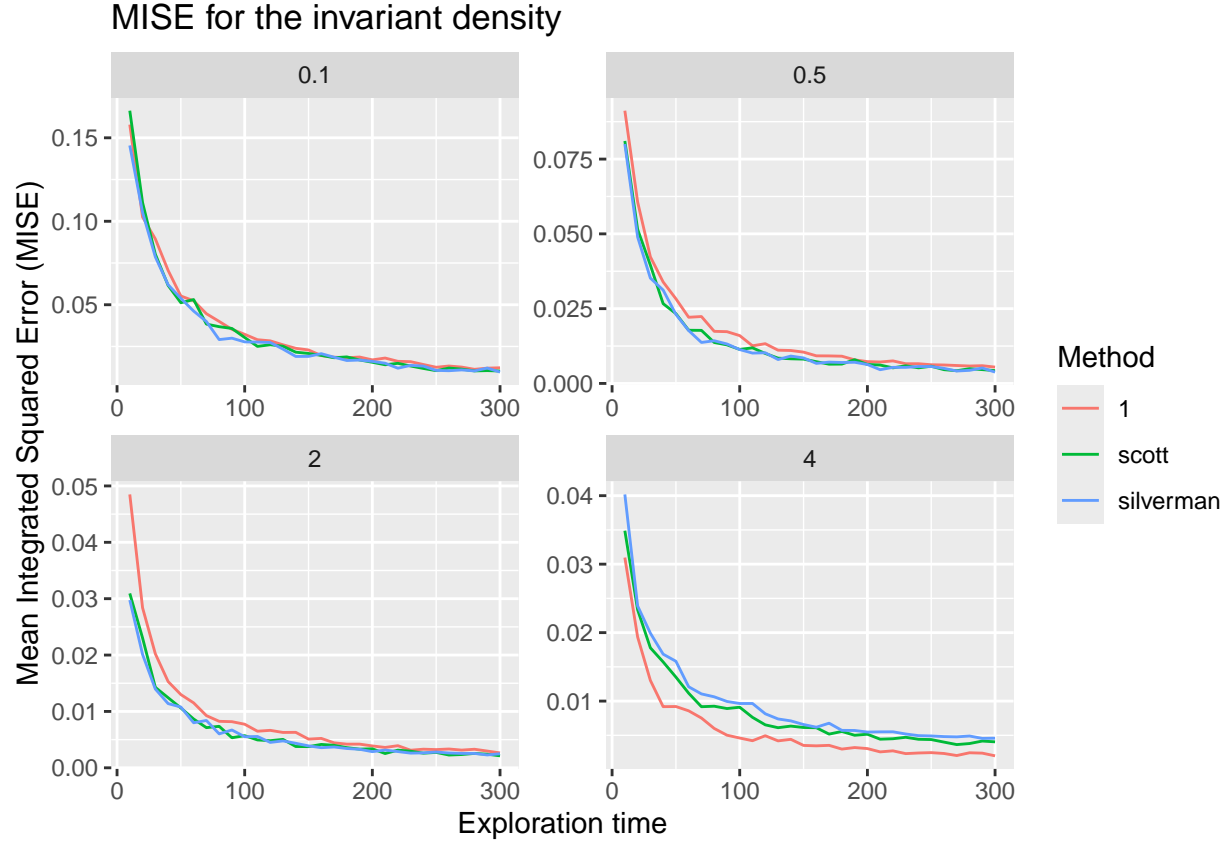
Standard deviation of Integrated Squared Error for the invariant density



We can see that the higher bandwidth powers have a slight advantage in making the estimations more robust in terms of the standard deviation of the integrated squared error for short exploration times for all drift slope coefficients. However, it does not take much exploration before all powers have the same stability for all the different drift slopes. Also, we see that the estimations become more stable when the drift slope coefficient increases.

Scott and silverman methods

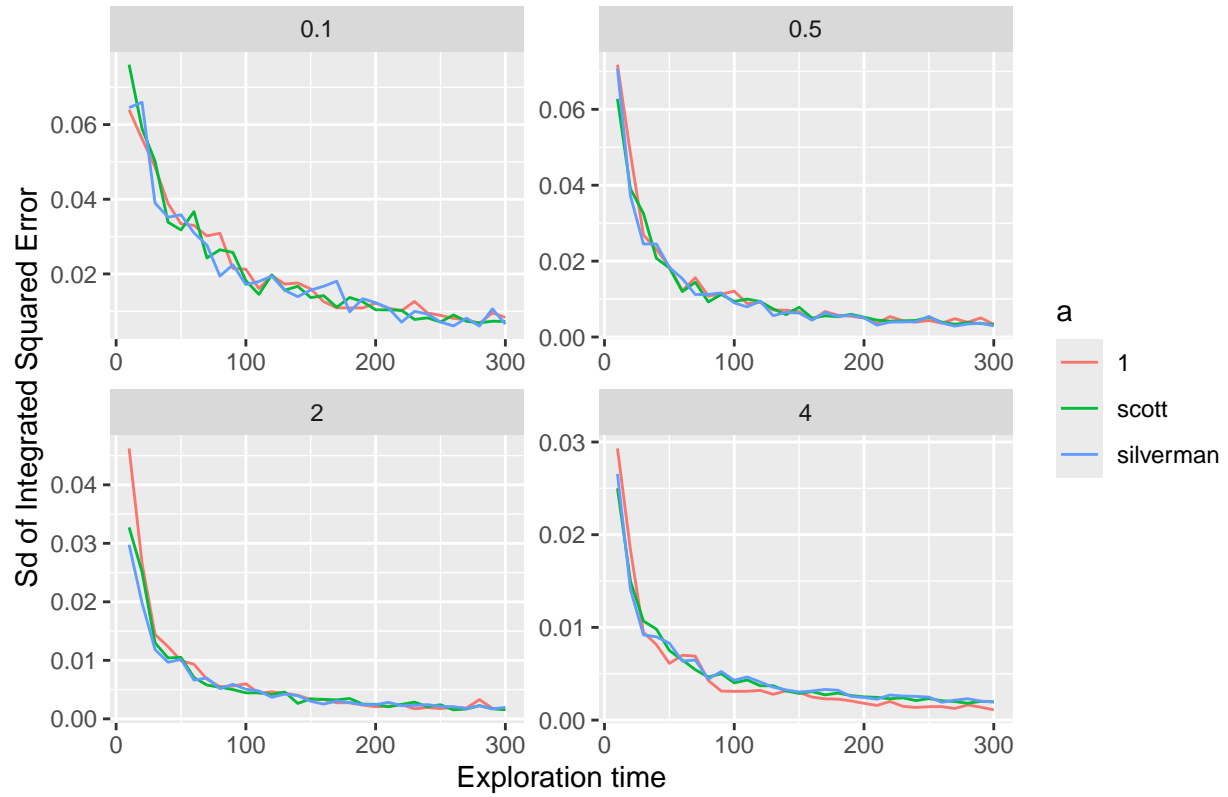
So far it seems that the best bandwidth is of the form $\frac{1}{\sqrt{T}}$, but as a final test we look at bandwidths that take the data that it is fitting on into consideration. Here we will look at the “scott” and “silverman” methods and compare them to the bandwidth of $\frac{1}{\sqrt{T}}$:



We can see that the “scott” and “silverman” methods only improve the MISE for the a drift slope coefficient of 2, and that the improvement is so small that it probably would be irrelevant to the data-driven algorithm. For the small drift slope coefficients we see that there is no difference in the MISE when using the “scott” or “silverman” methods, and for the drift slope coefficient of 4, we see that the bandwidth of $\frac{1}{\sqrt{T}}$ has a slight advantage through all of the exploration times, but again with such a small difference, that it would be irrelevant to the algorithm. Lastly, we see that the standard bandwidth and the “scott” and “silverman” methods all improve their fit, when the drift slope coefficient increases making the invariant distribution more concentrated around 0.

Lets look at the stability of these methods through the standard deviation of the integrated squared error:

Standard deviation of Integrated Squared Error for the invariant density



Again we see that using the “scott” or “silverman” method does not improve the stability of the kernel density estimation and that all three bandwidth methods improve their stability, when the drift slope coefficient increases.