

# Supplementary material

## S.1 The dynamic program

The Bayesian estimation needs the computation of the posterior probabilities of the variables involved, but the complexity of the problem does not allow us to find them analytically. Fortunately the computation can be done efficiently using dynamic programming. In this section we will briefly present the algorithm, while a more detailed explanation can be found in Hutter [11,12].

First of all, we introduce some notations:

$$\begin{aligned} \mathbf{M}_{l,m} &= (M_{l+1}, \dots, M_m) \\ \mathbf{T}_{l,m} &= (T_l, \dots, T_m) \\ \mathbf{Y}_{i,j} &= (Y_{i+1}, \dots, Y_j) \\ K_{i,j} &= \text{number of the segments in } \mathbf{Y}_{i,j}. \end{aligned}$$

The dynamic program is based on the fact that, for a fixed segment number and partition, the data points that belong to different segments are independent. Then, the joint probability distribution of the data points and the levels in  $m - l$  segments can be decomposed in the product of the same joint probability distribution in the first  $p - l$  segments and in the last  $m - p$  segments, for any  $p$  such that  $l < p \leq m$ :

$$\begin{aligned} &f(\mathbf{y}_{t_l, t_m}, \boldsymbol{\mu}_{l,m} | \mathbf{t}_{l,m}, K_{t_l, t_m} = m - l) \\ &= \prod_{i=l}^{m-1} f(\mathbf{y}_{t_i, t_{i+1}} | \mu_{i+1}, \mathbf{t}_{i, i+1}, K_{t_i, t_{i+1}} = 1) f(\mu_{i+1} | \mathbf{t}_{i, i+1}, K_{t_i, t_{i+1}} = 1) \\ &= f(\mathbf{y}_{t_l, t_p}, \boldsymbol{\mu}_{l,p} | \mathbf{t}_{l,p}, K_{t_l, t_p} = p - l) f(\mathbf{y}_{t_p, t_m}, \boldsymbol{\mu}_{p,m} | \mathbf{t}_{p,m}, K_{t_p, t_m} = m - p). \end{aligned} \quad (\text{S.1})$$

We define the following quantity to perform the dynamic program

$$A_{i,j}^r := \int_{\mathbb{R}} \mu_m^r f(\mathbf{y}_{i,j}, \mu_m | K_{i,j} = 1) d\mu_m \quad \text{for all } i < m \leq j, \quad (\text{S.2})$$

where  $\tilde{\mu}_{i+1} = \dots = \tilde{\mu}_j = \mu_m$ , since all the data points considered belong to the same segment, which we call  $m$ . One should notice that, if  $r = 0$ ,

$$A_{i,j}^0 = f(\mathbf{y}_{i,j} | K_{i,j} = 1), \quad (\text{S.3})$$

i.e., it is the density of the data  $\mathbf{y}_{i,j}$  given that they belong to the same segment, while if  $r \geq 1$ ,  $A_{i,j}^r$  is related to the moments of  $M_m$  with respect to the conditional probability given  $\mathbf{y}$ ,  $\mathbf{t}$  and  $k$ ,

$$E[M_m^r | \mathbf{y}, \mathbf{t}, k] = E[M_m^r | \mathbf{y}_{T_{m-1}, T_m}, T_{m-1} = i, T_m = j, K_{T_{m-1}, T_m} = 1] = \frac{A_{i,j}^r}{A_{i,j}^0} \quad (\text{S.4})$$

where  $i = t_{m-1}$  and  $j = t_m$ , for all  $m = 1, \dots, k$ .

The left and right recursion of the dynamic program are made on the following quantities

$$\mathbf{L}_{k+1,j} := \binom{j-1}{k} f(\mathbf{y}_{0,j} | K_{0,j} = k+1) \quad (\text{S.5})$$

$$\mathbf{R}_{k+1,i} := \binom{n-i-1}{k} f(\mathbf{y}_{i,n} | K_{i,n} = k+1). \quad (\text{S.6})$$

Given the position of  $t_k$ , the probability distribution of the data in  $k+1$  segments can be decomposed in the product between the joint distribution in the first  $k$  segments and the joint distribution in the last segment. Hence, we can compute the probability distribution of the data in  $k+1$  segments summing this products for all possible positions of  $t_k$ ,

$$\mathbf{L}_{k+1,j} = \sum_{h=k}^{j-1} \mathbf{L}_{k,h} \mathbf{A}_{h,j}^0. \quad (\text{S.7})$$

Similarly, we can sum, over all possible position of  $t_1$ , the products between the joint distribution in the first segment and the joint distribution in the last  $k$  segments, obtaining

$$\mathbf{R}_{k+1,i} = \sum_{h=i+1}^{n-k} \mathbf{A}_{i,h}^0 \mathbf{R}_{k,h}. \quad (\text{S.8})$$

The recursion starts with  $\mathbf{L}_{0,j} := \delta_{0,j}$  and  $\mathbf{R}_{0,i} := \delta_{i,n}$ , because there are zero segments only if the two endpoints are equal.

### The Bayesian estimators of the original BPCR.

For the definition of the Bayesian estimators, we need to compute the evidence and the posterior distributions of the parameters. Since

$$\mathbf{L}_{k,n} = \mathbf{R}_{k,0} = \binom{n-1}{k-1} f(\mathbf{y} | k), \quad (\text{S.9})$$

the *evidence* of the sample point  $\mathbf{y}$  turns out to be

$$\mathbf{E} := f(\mathbf{y}) = \sum_{k=1}^{k_{\max}} f(\mathbf{y} | k) p_K(k) = \frac{1}{k_{\max}} \sum_{k=1}^{k_{\max}} \frac{\mathbf{L}_{k,n}}{\binom{n-1}{k-1}}. \quad (\text{S.10})$$

Then, the posterior probabilities of the segment number and the boundaries can be written in the following way

$$\mathbf{C}_k := p(k | \mathbf{y}) = \frac{f(\mathbf{y} | k) p(k)}{f(\mathbf{y})} = \frac{\mathbf{L}_{k,n}}{\mathbf{E} k_{\max} \binom{n-1}{k-1}} \quad \forall k \in \mathbb{K} \quad (\text{S.11})$$

$$\mathbf{B}_{ph} := \mathbf{P}(T_p = h | \mathbf{y}, k_0) = \frac{f(T_p = h, \mathbf{y} | K = k_0)}{f(\mathbf{y} | k_0)} = \frac{\mathbf{L}_{p,h} \mathbf{R}_{k_0-p,h}}{\mathbf{L}_{k_0,n}} \quad (\text{S.12})$$

for each  $p = 1, \dots, k_0 - 1$  and for all  $h \in \{p, \dots, n-1\}$ .

Now we can compute the estimators. The MAP (Maximum A Posteriori) estimate of the segment number given the sample point  $\mathbf{y}$  is

$$\hat{k} := \arg \max_{k \in \mathbb{K}} p(k | \mathbf{y}) = \arg \max_{k \in \mathbb{K}} \frac{L_{k,n}}{\binom{n-1}{k-1}}. \quad (\text{S.13})$$

As we can see in Equation (S.12), the estimate for the boundaries needs the knowledge of the true number of segments, so we use  $\hat{k}$  instead of  $k_0$ ,

$$\hat{t}_p := \arg \max_{h \in \{p, \dots, n-(\hat{k}-p)\}} P(T_p = h | \mathbf{y}, \hat{k}) \quad \text{for } p = 1, \dots, \hat{k} - 1. \quad (\text{S.14})$$

The computation of the estimate of the  $r^{\text{th}}$  moment of the level of the  $m^{\text{th}}$  segment needs the knowledge of the segment number and the partition of the data (see Equation (S.4)), so we use the estimated ones

$$\hat{\mu}_m^r := E[M_m^r | \mathbf{y}, \hat{\mathbf{t}}, \hat{k}] = \frac{A_{i,j}^r}{A_{i,j}^0}, \quad (\text{S.15})$$

where  $i = \hat{t}_{m-1}$  and  $j = \hat{t}_m$ , for all  $m = 1, \dots, \hat{k}$ .

To estimate the segment level  $\tilde{M}_s$  at a generic position  $s$ , we need at first to compute explicitly its posterior distribution:

$$f(\tilde{\mu}_s | \mathbf{y}, K = k_0) = \sum_{m=1}^{k_0} \sum_{i=0}^{s-1} \sum_{j=s}^n f(\mu_m, T_{m-1} = i, T_m = j | \mathbf{y}, K = k_0) \quad (\text{S.16})$$

$$= \frac{1}{L_{k_0,n}} \sum_{m=1}^{k_0} \sum_{i=0}^{s-1} \sum_{j=s}^n L_{m-1,i} f(\mathbf{y}_{i,j}, \mu_m | K_{i,j} = 1) R_{k_0-m,j}. \quad (\text{S.17})$$

Hence the estimate of  $\tilde{M}_s$  given  $\hat{k}$  is

$$\hat{\mu}_s := \sum_{i=0}^{s-1} \sum_{j=s}^n F_{i,j}^0 \quad \text{with } F_{i,j}^0 := \frac{1}{L_{\hat{k},n}} \sum_{m=1}^{\hat{k}} L_{m-1,i} A_{i,j}^0 R_{\hat{k}-m,j}, \quad (\text{S.18})$$

for all  $s = 1, \dots, n$ .

**The boundary estimator  $\hat{\mathbf{T}}_{\text{joint}}$ .**

The boundary estimator  $\hat{\mathbf{T}}_{\text{joint}}$  is defined as

$$\hat{\mathbf{T}}_{\text{joint}} := \arg \max_{\mathbf{t} \in \mathbb{T}_{k_0,n}} p(\mathbf{t} | K = k_0, \mathbf{Y}). \quad (\text{S.19})$$

The explicit formula for the joint boundary distribution, given  $k_0$  and the sample point  $\mathbf{y}$ , is

$$\begin{aligned} p(\mathbf{t} | \mathbf{y}, K = k_0) &= \frac{\left[ \prod_{p=0}^{k_0-1} f(\mathbf{y}_{t_p, t_{p+1}} | t_p, t_{p+1}, K_{t_p, t_{p+1}} = 1) \right] p(\mathbf{t} | K = k_0)}{f(\mathbf{y} | K = k_0)} \\ &= \frac{\left[ \prod_{p=0}^{k_0-1} A_{t_p, t_{p+1}}^0 \right]}{L_{k_0,n}}. \end{aligned} \quad (\text{S.20})$$

Thus, the estimated boundaries are

$$\widehat{\mathbf{t}}_{\text{joint}} = \arg \max_{\mathbf{t} \in \mathbb{T}_{k_0, n}} \prod_{p=0}^{k_0-1} A_{t_p, t_{p+1}}^0. \quad (\text{S.21})$$

This estimator can be computed by using the following recursion,

$$\begin{aligned} W_{0,i}^1 &:= A_{0,i}^0 \\ W_{0,i}^{p+1} &:= \max_{h \in \{p, \dots, i-1\}} W_{0,h}^p A_{h,i}^0 \quad \text{for } p = 1, \dots, \widehat{k} - 2, \end{aligned} \quad (\text{S.22})$$

where  $W_{0,i}^p$  represents the maximum probability that  $\mathbf{y}_{0,i}$  is divided into  $p$  segments over all combinations of the boundaries. As a consequence, the components of the vector of the boundary estimator turn out to be,

$$\widehat{t}_k = \arg \max_{h \in \{k, \dots, \widehat{t}_{k+1}-1\}} W_{0,h}^k A_{h, \widehat{t}_{k+1}}^0 \quad \text{for } k = \widehat{k} - 1, \dots, 1, \quad (\text{S.23})$$

with  $\widehat{t}_{\widehat{k}} := n$ .

**The boundary estimator  $\widehat{\mathbf{T}}_{\text{BinErrAk}}$ .**

To compute the  $\widehat{\mathbf{T}}_{\text{BinErrAk}}$  estimator, we need at first to make explicit the expected value in Equation (22), given the sample point  $\mathbf{y}$ ,

$$\begin{aligned} \mathbb{E} \left[ \sum_{i=1}^{n-1} \mathcal{T}_i \tau'_i \middle| \mathbf{y} \right] &= \sum_{k=2}^{k_{\max}} \sum_{\boldsymbol{\tau} \in \mathcal{T}_{k,n}} \sum_{i=1}^{n-1} \tau_i \tau'_i p(\boldsymbol{\tau} | \mathbf{y}, k) p(k | \mathbf{y}) \\ &= \sum_{i=1}^{n-1} \delta_{\tau'_i, 1} \sum_{k=2}^{k_{\max}} \sum_{p=1}^{\min(i, k-1)} \frac{L_{p,i} R_{k-p,i}}{L_{k,n}} \frac{L_{k,n}}{\binom{n-1}{k-1} k_{\max} \mathbb{E}} \end{aligned} \quad (\text{S.24})$$

$$= \sum_{\{i=1, \dots, n-1 : \tau'_i=1\}} \mathbb{P}(\tau_i = 1 | \mathbf{Y} = \mathbf{y}), \quad (\text{S.25})$$

where

$$\begin{aligned} \mathbb{P}(\tau_i = 1 | \mathbf{Y} = \mathbf{y}) &= \sum_{k=2}^{k_{\max}} \mathbb{P}(\mathbf{t} \in \mathbb{T}_{k,n} : \exists p \text{ such that } t_p = i | \mathbf{y}, k) p(k | \mathbf{y}) \\ &= \sum_{k=2}^{k_{\max}} \sum_{p=1}^{\min(i, k-1)} \mathbb{P}(\mathbf{t} \in \mathbb{T}_{k,n} : t_p = i | \mathbf{y}, k) p(k | \mathbf{y}). \end{aligned} \quad (\text{S.26})$$

Hence, to find the estimated  $\widehat{\boldsymbol{\tau}}_{\text{BinErrAk}}$ , we need to find the  $\widehat{k} - 1$  highest  $\{\mathbb{P}(\tau_i = 1 | \mathbf{Y} = \mathbf{y})\}_{i=1}^{n-1}$  (corresponding to the indices  $i_1, \dots, i_{\widehat{k}-1}$ ) and then take  $\widehat{\boldsymbol{\tau}}_{\text{BinErrAk}}$  such that  $\widehat{\tau}_{i_p} = 1$  for  $p = 1, \dots, \widehat{k} - 1$ . Using the inverse function of (19), we obtain the corresponding value of the estimator  $\widehat{\mathbf{T}}_{\text{BinErrAk}}$ .

**The BRCAk.**

The computation of the BRCAk is simply given by

$$\begin{aligned}
\widehat{M}_s &:= E[\widetilde{M}_s | \mathbf{y}] \\
&= \sum_{k=1}^{k_{\max}} p(k | \mathbf{y}) \int_{\mathbb{R}} \widetilde{\mu}_s f(\widetilde{\mu}_s | \mathbf{y}, k) d\widetilde{\mu}_s \\
&= \sum_{k=1}^{k_{\max}} C_k \sum_{i < t \leq j} F_{i,j}^0(k),
\end{aligned} \tag{S.27}$$

where in the last equation we used the notations of Hutter [11,12] with the only difference that now  $F_{i,j}^0$  must be computed for all possible  $k$  and so we need to make explicit this dependency. Unfortunately, the computation of this quantity required time  $O(n^2 k_{\max}^2)$ , hence it could be a problem with samples of big size.

## S.2 Some error measure definitions

### Definition of the MSE measure.

The mean square error is defined as the expected value of the square error and so it is estimated with the corresponding arithmetic mean. Hence, given  $N$  estimated values  $\hat{\theta}_1, \dots, \hat{\theta}_N$  of a generic parameter  $\theta$ , the estimated Mean Square Error of  $\theta$  is defined as

$$\text{MSE}(\theta) := \frac{1}{N} \sum_{i=1}^N (\hat{\theta}_i - \theta)^2.$$

### Definition of the mean relative error.

Given  $N$  estimated values  $\hat{\theta}_1, \dots, \hat{\theta}_N$  of a generic parameter  $\theta$ , the estimated mean relative error over all datasets is

$$\frac{\Delta \hat{\theta}}{\theta} := \frac{1}{mn} \sum_{j=1}^m \sum_{i=1}^n \frac{|\hat{\theta}_{ij} - \theta_j|}{\theta_j},$$

where  $m$  is the number of datasets,  $n$  the number of samples in each dataset and  $\theta_{ij}$  is the estimate of  $\theta_j$  based on the  $i^{\text{th}}$  sample in the  $j^{\text{th}}$  dataset.

## S.3 Supplementary Tables

$\Delta \hat{\nu} / \nu$	0.6376
$\Delta \hat{\sigma}^2 / \sigma^2$	0.1524
$\Delta \hat{\rho}^2 / \rho^2$	8.6217
$\Delta \hat{\rho}_1^2 / \rho_1^2$	0.5840

Table S.1: Estimated mean relative error of the estimators  $\hat{\nu}$ ,  $\hat{\sigma}^2$ ,  $\hat{\rho}^2$  and  $\hat{\rho}_1^2$  over all the datasets used. The results show that  $\hat{\sigma}^2$  had the lowest error. The error of  $\hat{\nu}$  was high but acceptable. The estimator  $\hat{\rho}_1^2$  was significantly better than  $\hat{\rho}^2$  with respect to this error measure.

error \ $\hat{T}$	$\hat{T}_{01}$	$\hat{T}_{joint}$	$\hat{T}_{BinErr}$	$\hat{T}_{BinErrAk}$
sum 0-1	14.37 ± 0.176	16.04 ± 0.195	16 ± 0.194	15.99 ± 0.193
joint 0-1	0.9393 ± 0.00436	0.945 ± 0.0042	0.9387 ± 0.00438	0.9387 ± 0.00438
binary	10.48 ± 0.126	8.3787 ± 0.0961	7.7433 ± 0.09	7.7793 ± 0.0903
aver. square	35.12 ± 15.667	67.3 ± 17.036	37.85 ± 15.792	38.76 ± 15.824

Table S.2: Estimated mean errors ± standard deviation of the estimators of  $t^0$  applied on a dataset without replicates with  $\sigma^2 = 0.1$  and  $\rho^2 = 0.5$ , using  $k_0$ ,  $\hat{\nu}$ ,  $\hat{\sigma}^2$  and  $\hat{\rho}_1^2$ . The estimator  $\hat{T}_{01}$  minimized the sum 0-1 error and the average square error. The errors of  $\hat{T}_{BinErr}$  and  $\hat{T}_{BinErrAk}$  were always very close and low.

gene region	FISH CN	mBPCR		CBS	CGHseg	HMM	GLAD	BioHMM	Rendersome
		$\hat{\rho}^2$	$\hat{\rho}_1^2$						
<i>BCL6</i>	2/3	2.89	2.85	2.89	2.90	2.05	2.79	2.85	2.86
<i>C-MYC</i>	2	2.02	1.99	2.06	2.12	2.05	2.07	2.02	2.02
<i>CCND1</i>	2	2.01	1.92	2.01	2.05	2.05	2.07	2.02	2.02
<i>BIRC3</i>	2/3	2.01	2.22	2.01	2.05	2.05	2.07	2.02	2.02
<i>ATM</i>	2	2.01	1.80	2.01	2.05	2.05	2.07	2.02	2.02
<i>D13S319</i>	2	2.03	2.40	1.98	2.03	2.05	2.01	2.02	2.89
<i>LAMP1</i>	2	1.82	1.76	1.98	1.98	2.05	2.01	2.02	2.02
<i>TP53</i>	1/2	1.11	1.17	1.11	1.11	1.10	1.55	1.10	1.20
<i>MALT1</i>	2/3	2.12	2.25	2.02	2.12	2.05	2.09	2.02	2.02
<i>BCL2</i>	2	2.12	2.25	2.02	2.12	2.05	2.09	2.02	2.02

Table S.3: Copy number estimation results obtained on sample REC-1. Globally, all methods behaved equally well on this data. Only Rendersome was unable to detect the correct copy number of *D13S319*.

gene region	FISH CN	mBPCR		CBS	CGHseg	HMM	GLAD	BioHMM	Renders.
		$\hat{\rho}^2$	$\hat{\rho}_1^2$						
<i>BCL6</i>	2	2.11	2.10	2.11	2.07	2.11	1.85	2.12	2.04
<i>C-MYC</i>	2	2.07	2.00	2.08	2.11	1.99	6.22/1.37	2.12	2.04
<i>CCND1</i>	2	2.06	2.03	2.06	2.34	2.20	2.4	2.12	2.04
<i>BIRC3</i>	2/3	2.06	1.76	2.06	1.14/2.34	2.11	2.4	2.12	2.04
<i>ATM</i>	1	2.06	2.01/1.61	2.06	2.34	2.11/1.14	2.4	2.12	2.04
<i>D13S319</i>	2	2.01	2.00	2.03	2.05	2.07	2.26	2.12	2.04
<i>LAMP1</i>	2	2.01	2.00	2.03	2.05	1.98	1.58	2.12	2.04
<i>TP53</i>	1	1.10	1.16	1.13	1.01	1.13	1.85	1.36	1.08
<i>MALT1</i>	3	3.36	3.05	3.17	3.04	5.30	2.16	4.78	4.28
<i>BCL2</i>	ampl	5.46	5.10	5.52	6.12	5.30	2.16	4.78	7.22

Table S.4: Copy number estimation results obtained on sample GRANTA-519. On this data, the GLAD method often did not detect the right gene copy number. The method mBPCR with  $\hat{\rho}_1^2$  estimated the gene copy number always well, apart from *ATM* whose copy number was estimated different from the FISH copy number by all methods.

gene region	FISH CN	mBPCR		CBS	CGHseg	HMM	GLAD	BioHMM	Rendersome
		$\hat{\rho}^2$	$\hat{\rho}_1^2$						
<i>BCL6</i>	3/2	3.06	3.06	3.02	3.06	2.96	2.98	3.04	2.98
<i>C-MYC</i>	ampl	7.12	7.10	7.14	6.87	6.70/2.63	7.28	6.51	7.72
<i>CCND1</i>	2	3.51	3.51	3.51	3.51	3.44	3.51	3.52	3.60
<i>BIRC3</i>	4/5	4.20	4.19	4.20	4.24	4.24	4.23	4.26	3.60
<i>ATM</i>	4	4.20	4.19	4.20	4.24	4.24	4.23	4.26	3.60
<i>D13S319</i>	4	3.72	3.72	3.81	3.82	3.72	3.81	3.73	3.64
<i>LAMP1</i>	4	3.67	3.67	3.82	3.67	3.72	3.69	3.73	3.64
<i>TP53</i>	2/3	2.57	2.69	2.22	2.76	2.34	2.76	2.83	2.90
<i>MALT1</i>	4	3.52	3.52	3.59	3.59	3.50	3.55	3.52	3.50
<i>BCL2</i>	4	3.52	3.52	3.59	3.59	3.50	3.55	3.52	3.50

Table S.5: Copy number estimation results obtained on 250K Array data of sample JEKO-1. All methods behaved equally good. The method HMM had problem in determining the right position of one breakpoint around the *C-MYC* amplification. All methods estimated the copy number of *CCND1* differently from FISH technique.

## S.4 Supplementary Figures

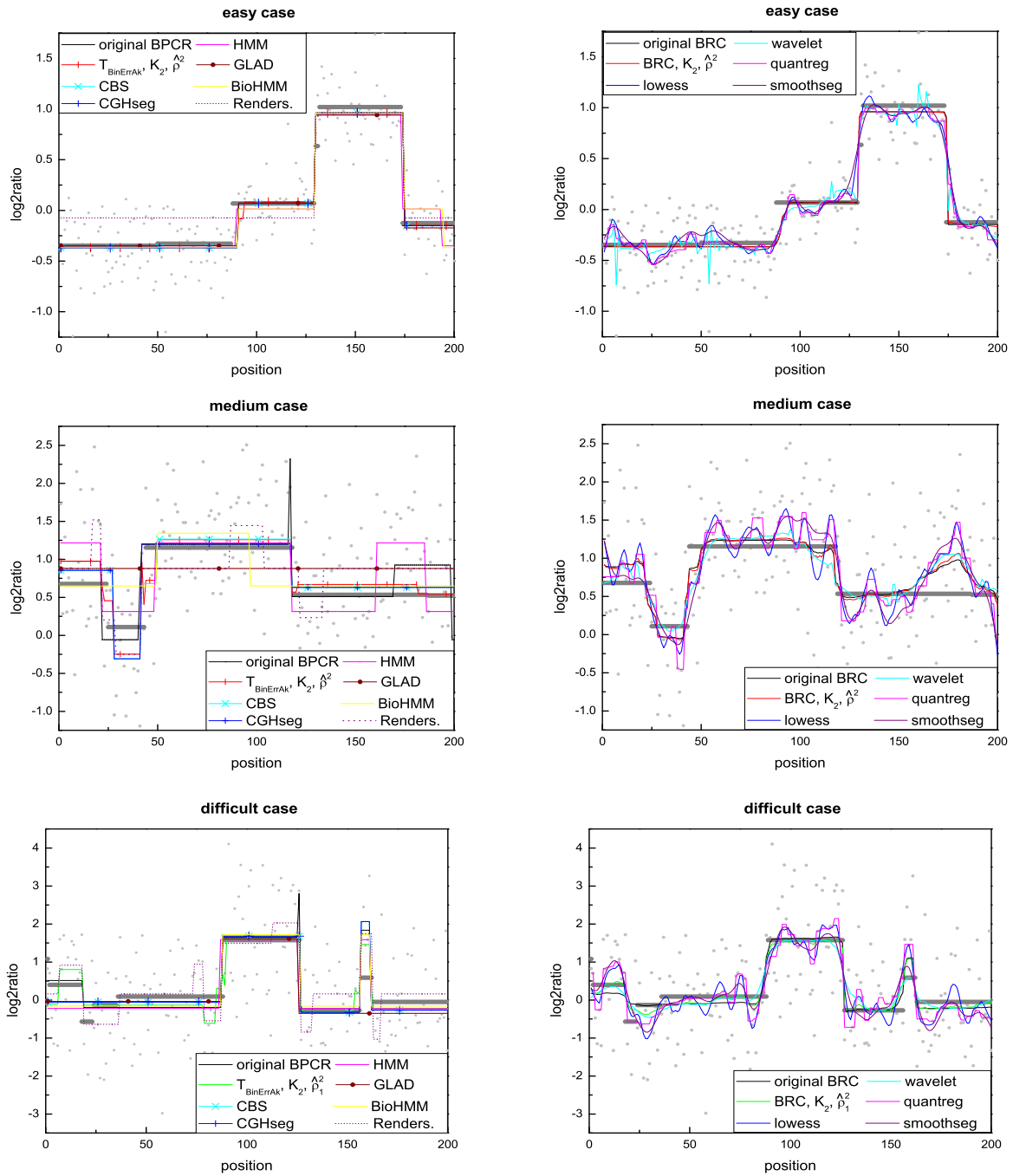


Figure S.1: Estimated profiles of the data shown in Figure 1, obtained by applying several methods. In each plot, the grey segments represent the true profile and the dots are the raw data points.

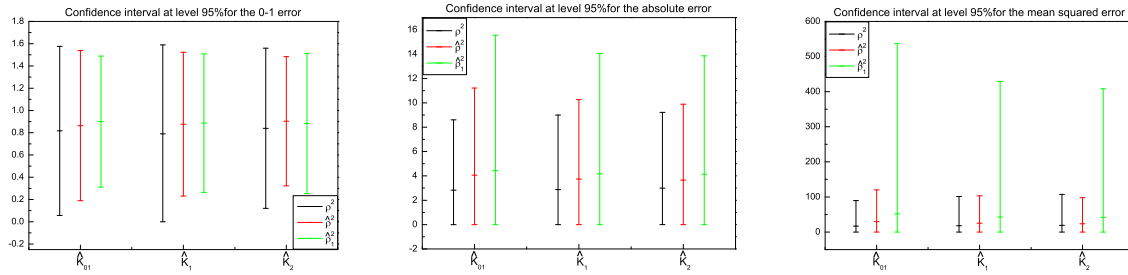


Figure S.2: Confidence intervals at 95% of the estimators of  $k_0$  for, respectively, the 0-1 error, the absolute error and the squared error obtained on a dataset without replicates with  $\sigma^2 = 0.1$  and  $\rho^2 = 0.5$ . The graphs show that, in each situation,  $\widehat{K}_2$  always had the lowest upper bound of the interval. Using  $\widehat{\rho}^2$  the confidence intervals were shorter.

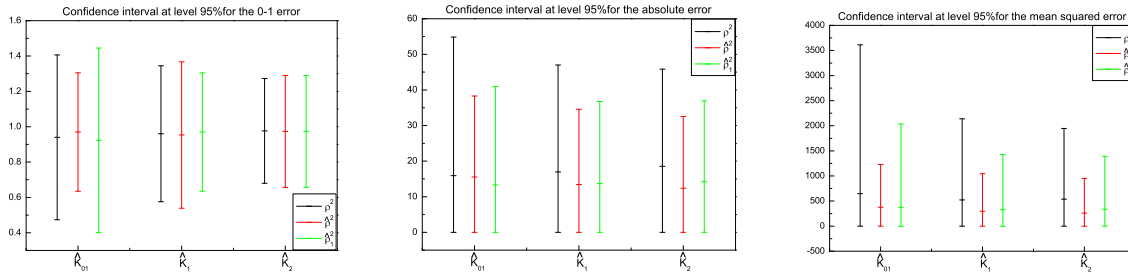


Figure S.3: Confidence intervals at 95% of the estimators of  $k_0$  for, respectively, the 0-1 error, the absolute error and the squared error obtained on a dataset without replicates with  $\sigma^2 = 0.3$  and  $\rho^2 = 0.05$ . The graphs show that, in each situation,  $\widehat{K}_2$  always had the lowest upper bound of the interval.

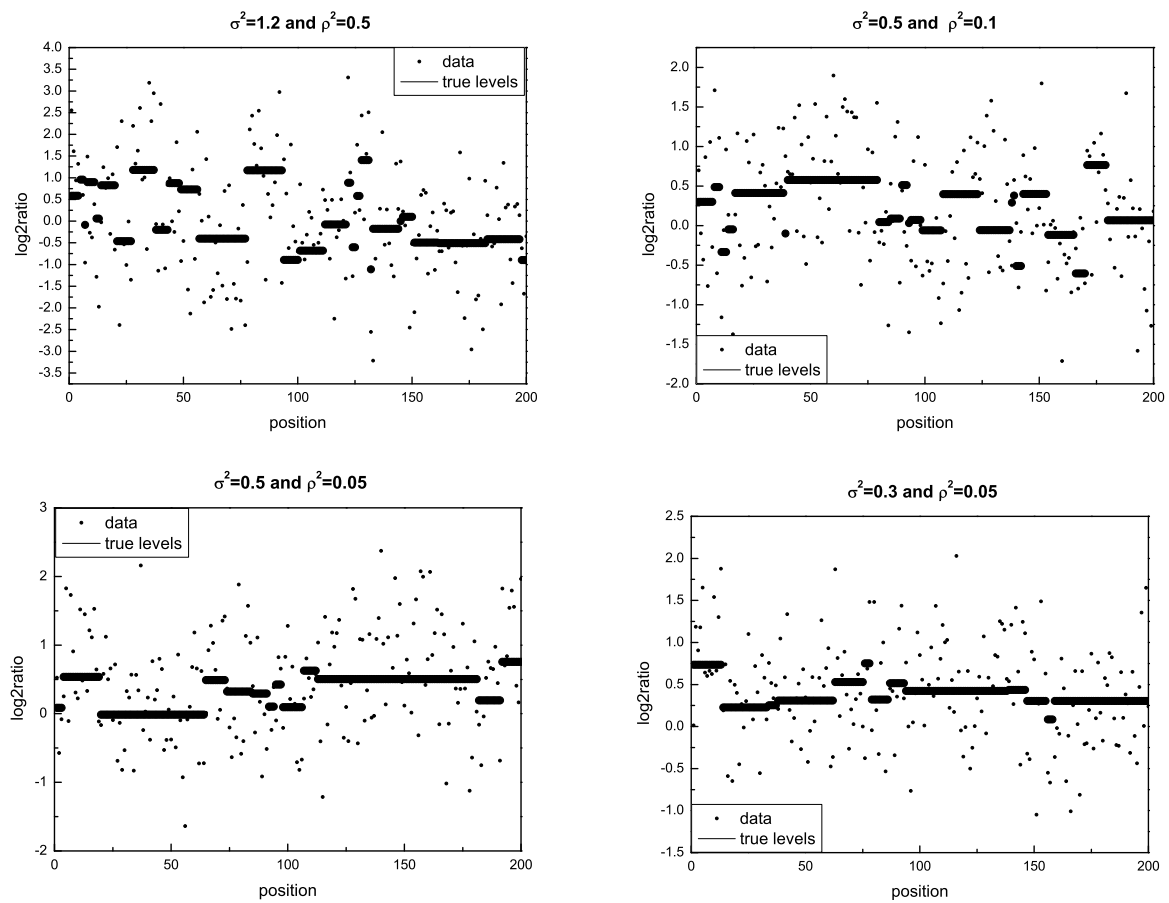


Figure S.4: True profiles of some datasets with replicates used in the comparisons.

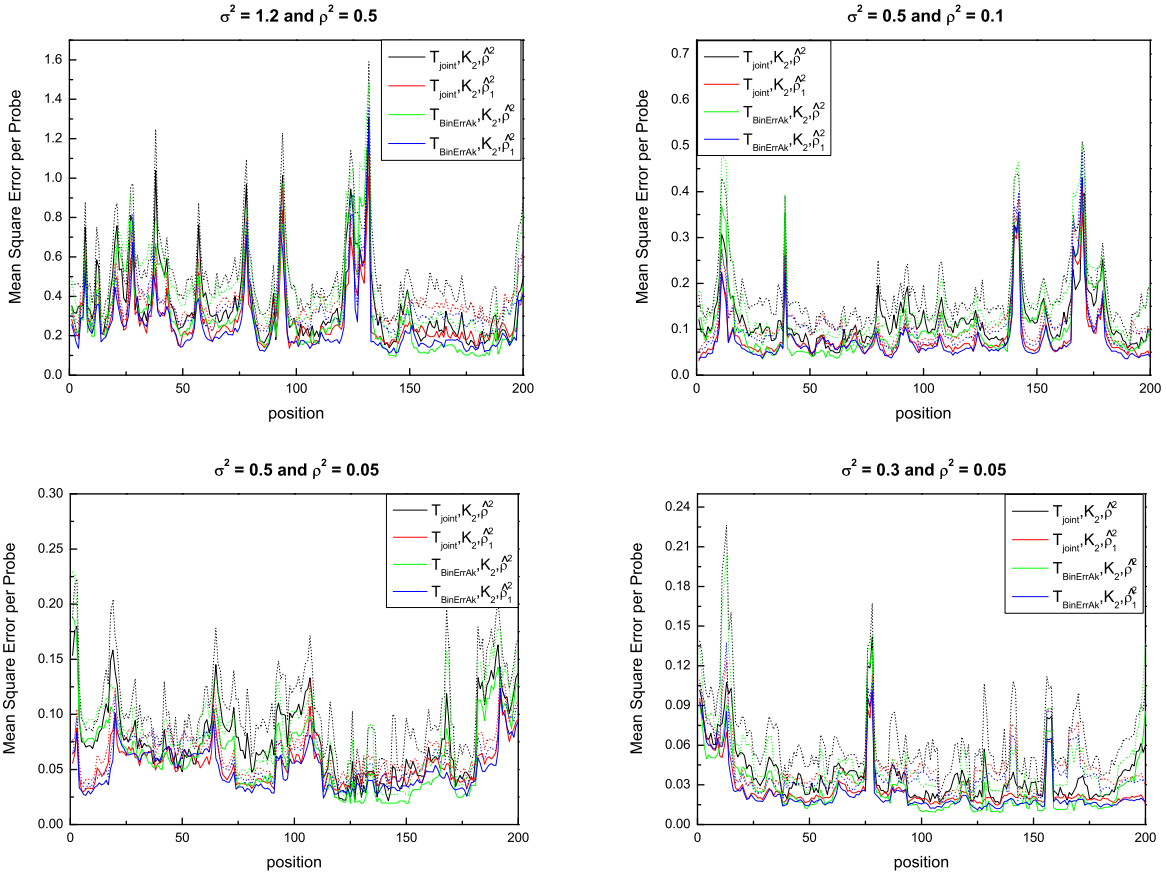


Figure S.5: Comparison of the segment level estimation by using  $\hat{T}_{joint}$  or  $\hat{T}_{BinErrAk}$  with the different  $\rho^2$  estimators, on four datasets with replicates. The corresponding true profiles are in Figure S.4. In general, using  $\hat{T}_{BinErrAk}$  we obtained a lower MSE per probe than using  $\hat{T}_{joint}$ . For a fixed boundary estimator, often the error was lower by using  $\hat{\rho}_1^2$  on these datasets where  $\sigma^2 \gg \rho^2$ .

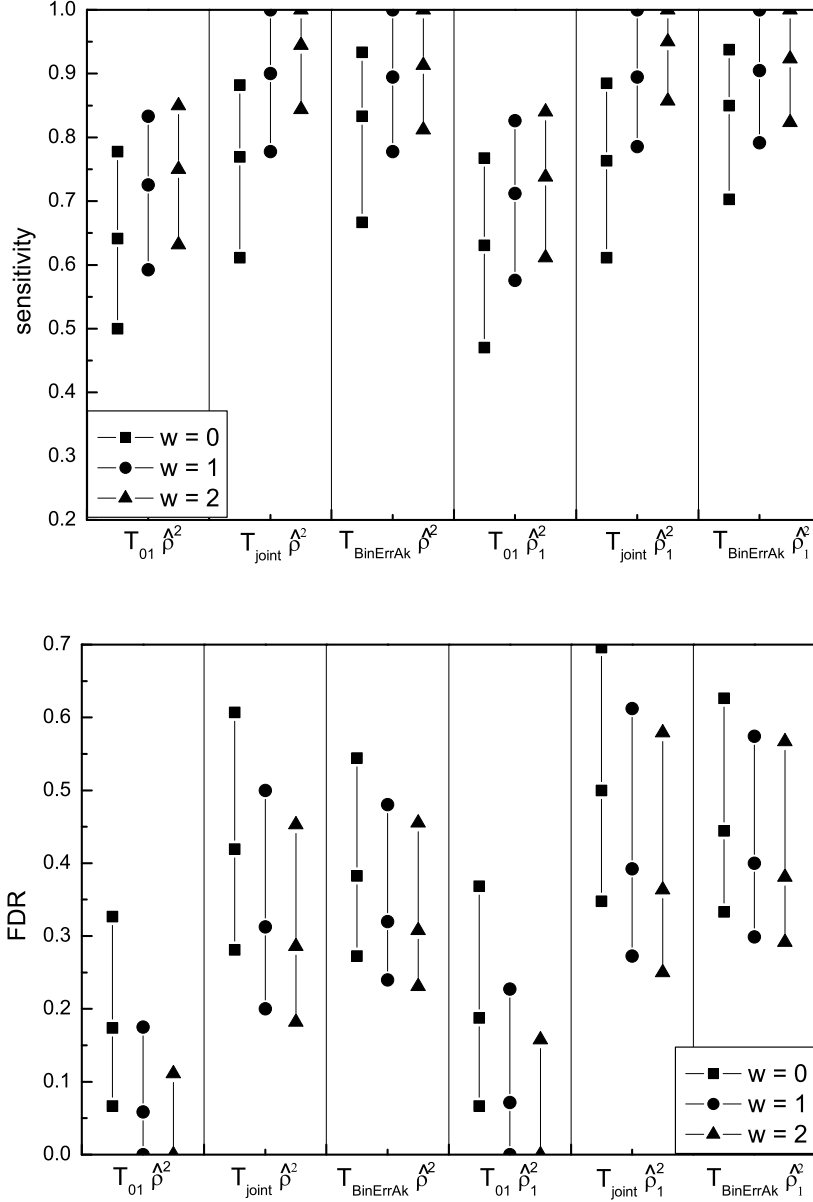


Figure S.6: Comparison of the sensitivity and FDR computed on the results obtained on the dataset *Simulated Chromosomes* using all the piecewise constant methods. Using  $\hat{\rho}^2$  instead of  $\hat{\rho}_1^2$ , the FDR was lower. The estimator  $\hat{T}_{BinErrAk}$  had the highest sensitivity and the second lowest FDR. The FDR of  $\hat{T}_{01}$  was the lowest, but this is due to the fact that it reduces the number of the estimated breakpoints by assigning the same position to different breakpoints. In conclusion,  $\hat{T}_{BinErrAk}$  with  $\hat{\rho}^2$  was the best performing estimator on this dataset.

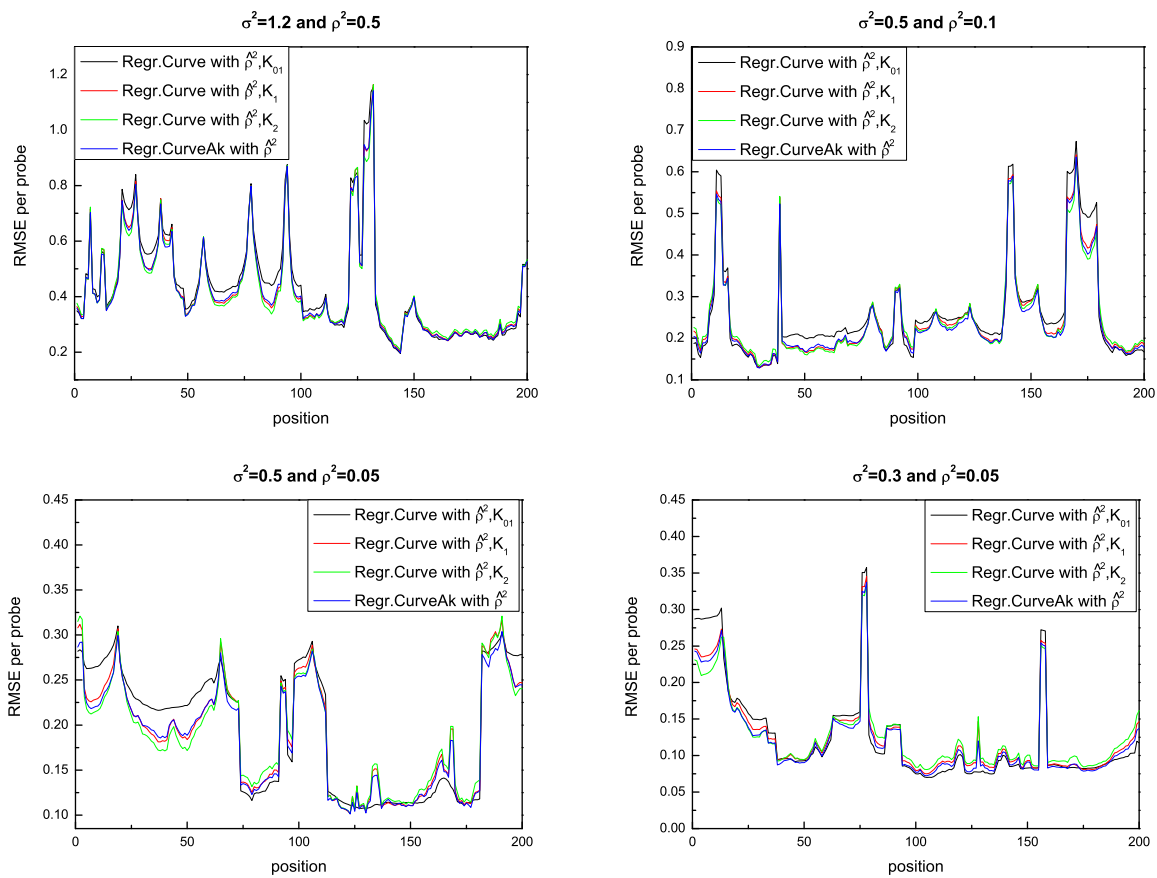


Figure S.7: RMSE per probe of the several Bayesian regression curves, using  $\hat{\rho}^2$  as the estimator of  $\rho^2$ , on four datasets with replicates. The corresponding true profiles are in Figure S.4. The graphs show that, using  $\hat{\rho}^2$ , BRCAk always had the lowest RMSE per probe and thus performed better than the other BRCs.

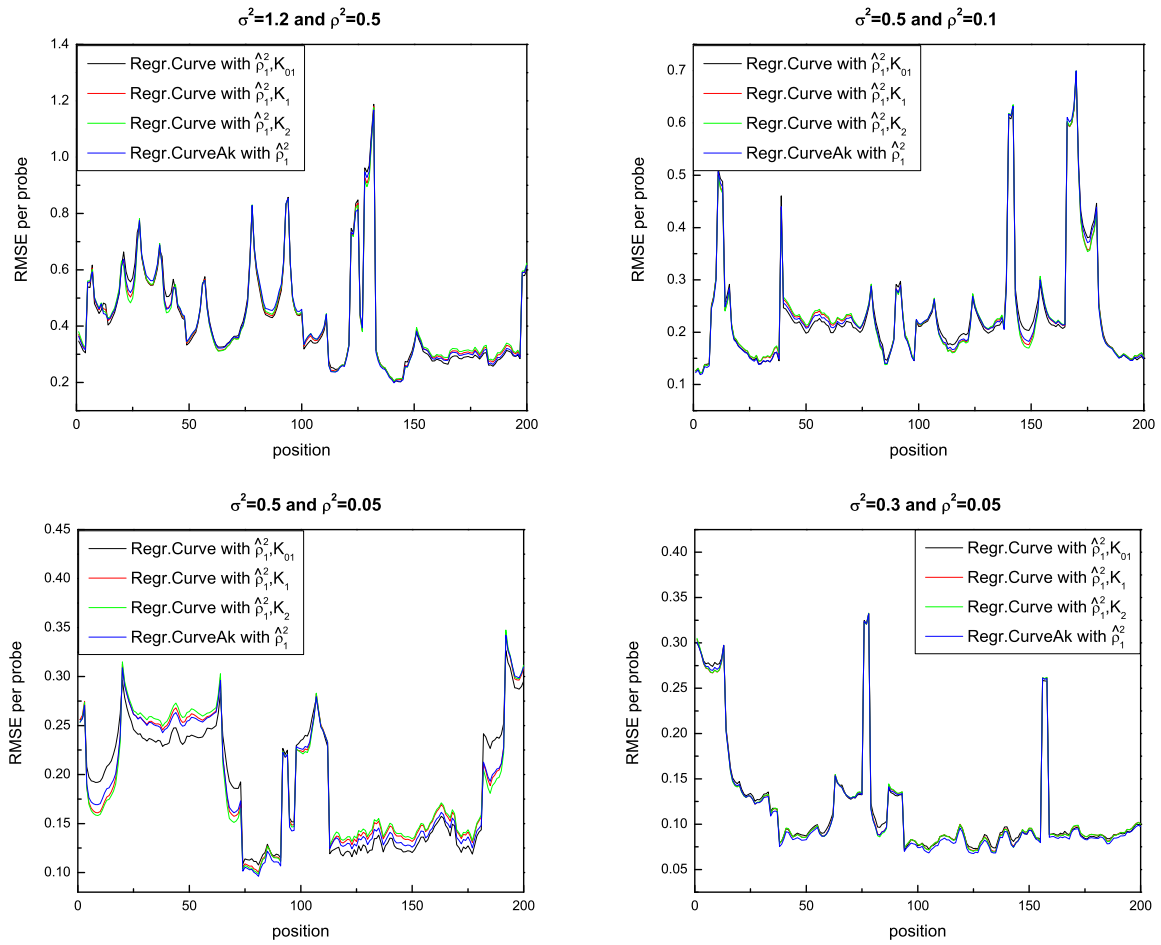


Figure S.8: RMSE per probe of the several Bayesian regression curves, using  $\hat{\rho}_1^2$  as the estimator of  $\rho^2$ , on four datasets with replicates. The corresponding true profiles are in Figure S.4. Using the estimator  $\hat{\rho}_1^2$ , all the regression curves gave similar RMSE per probe curve.

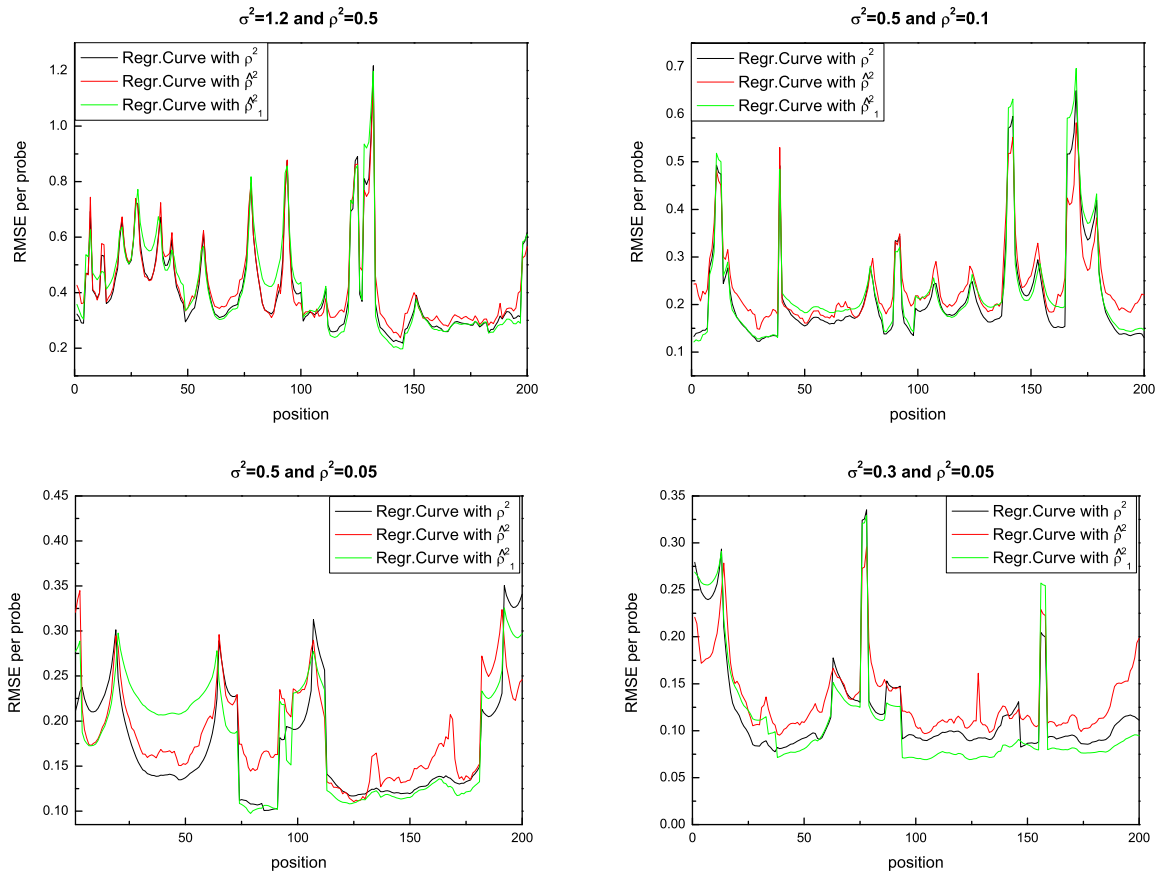


Figure S.9: RMSE per probe of the Bayesian regression curve using different estimators of  $\rho^2$ , on four datasets with replicates. The corresponding true profiles are in Figure S.4. The graphs do not show clearly which  $\rho^2$  estimator was better with respect to this error measure. Sometimes the error committed using  $\hat{\rho}_1^2$  was lower than using  $\hat{\rho}^2$ , probably because  $\hat{\rho}_1^2$  can lead to a slight overfitting.

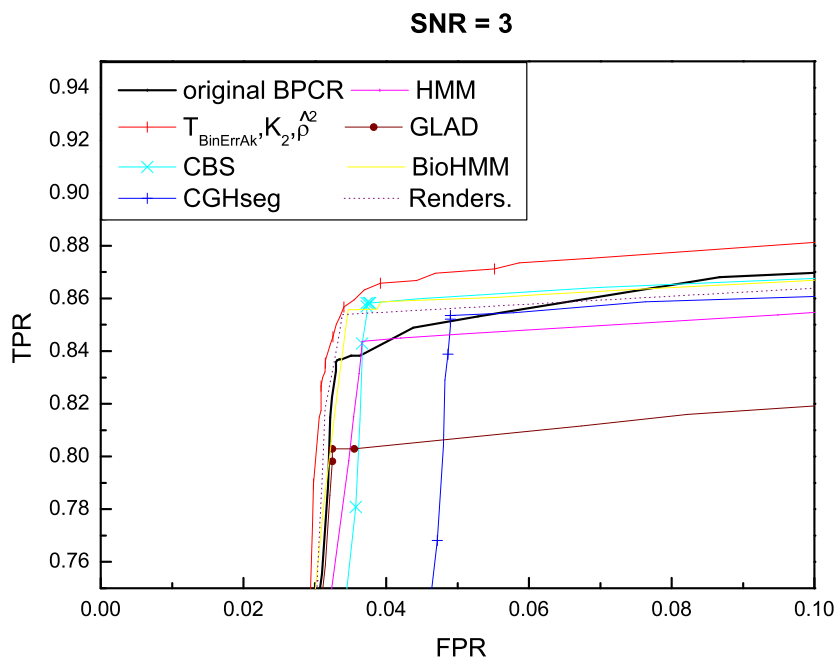
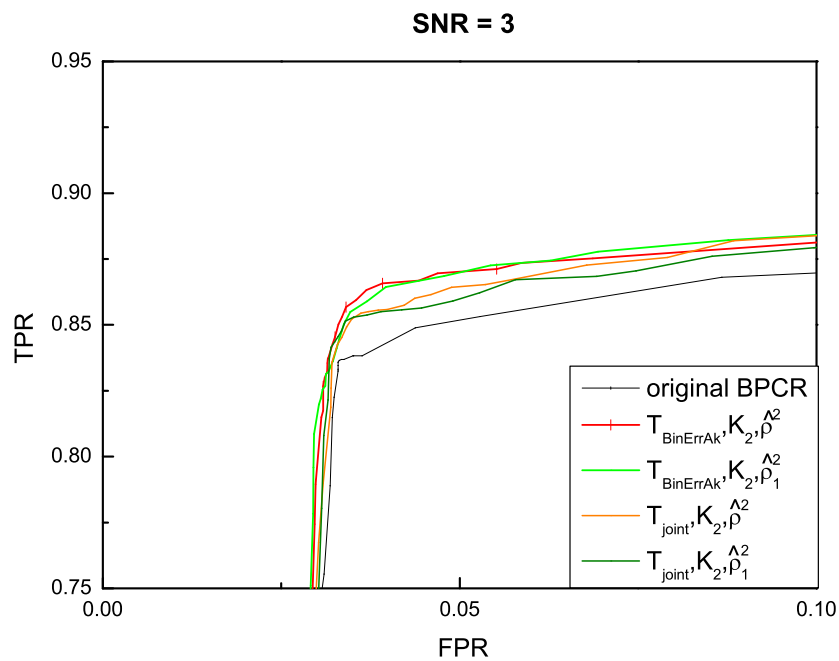


Figure S.10: Zoomed ROC curves of several piecewise constant methods applied to dataset with  $\text{SNR} = 3$ . On this easy type of data, all the methods (apart from GLAD) performed well, since their ROC curves were close to the top left corner of the plot.

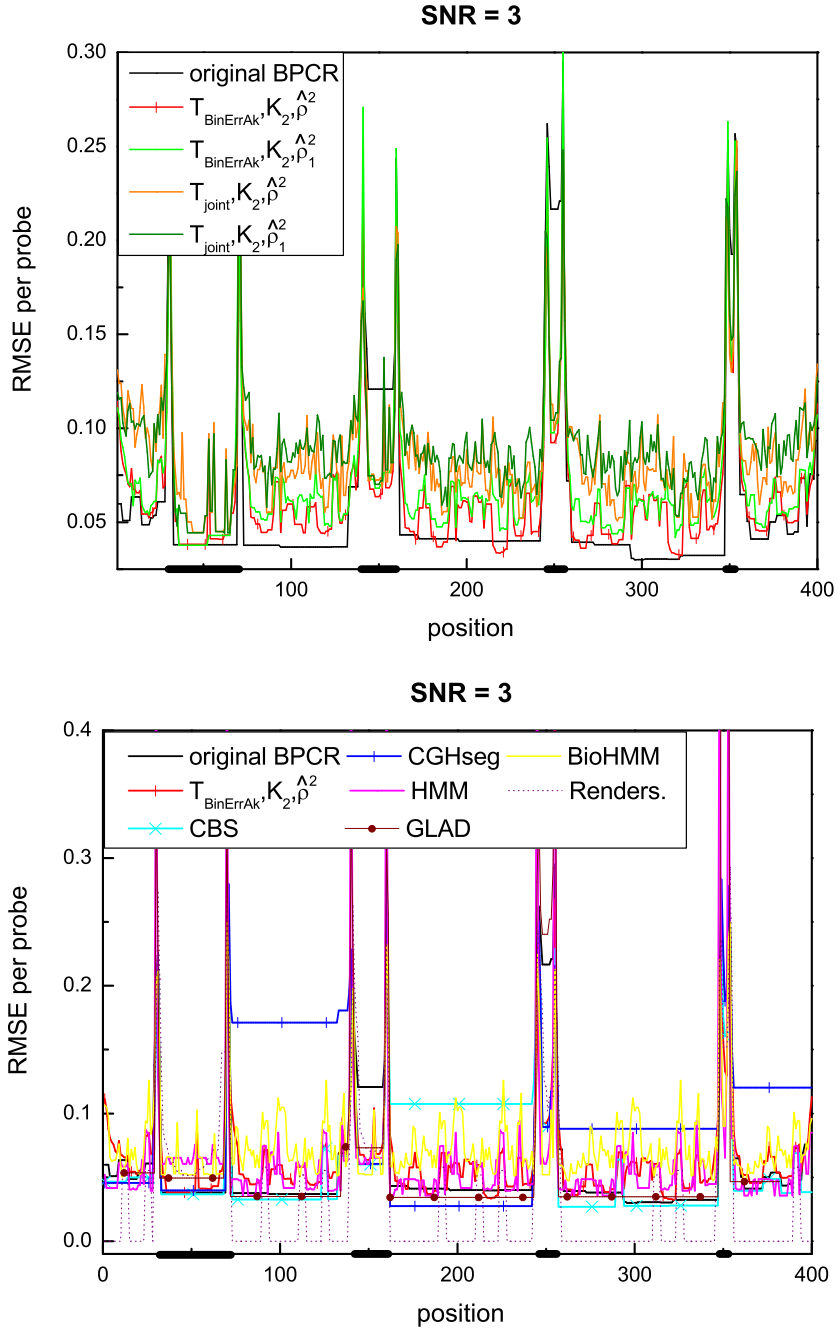


Figure S.11: RMSE of several piecewise constant methods applied to dataset with  $SNR = 3$ . The black segments on the horizontal axis correspond to the regions of aberration. On this dataset, HMM and the modified BPCR with  $\hat{\rho}^2$ ,  $\hat{K}_2$  and  $\hat{T}_{BinErrAk}$  achieved a low RMSE both inside and outside the aberrations, compared to the other algorithms. Nevertheless, HMM had the highest error at the breakpoint positions. Hence, the modified BPCR with  $\hat{\rho}^2$ ,  $\hat{K}_2$  and  $\hat{T}_{BinErrAk}$  performed better than all other methods with respect to the RMSE per probe measure.

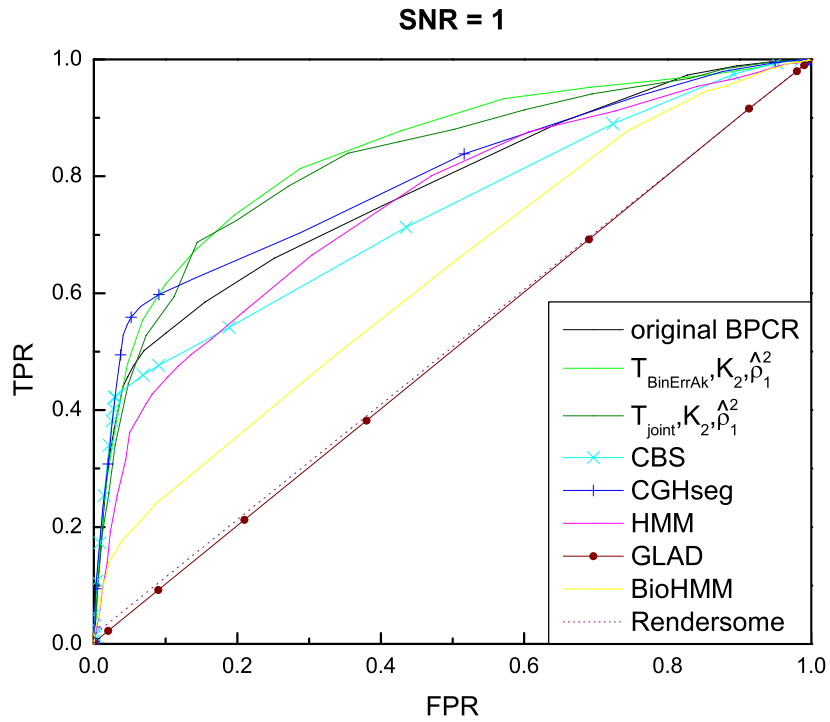
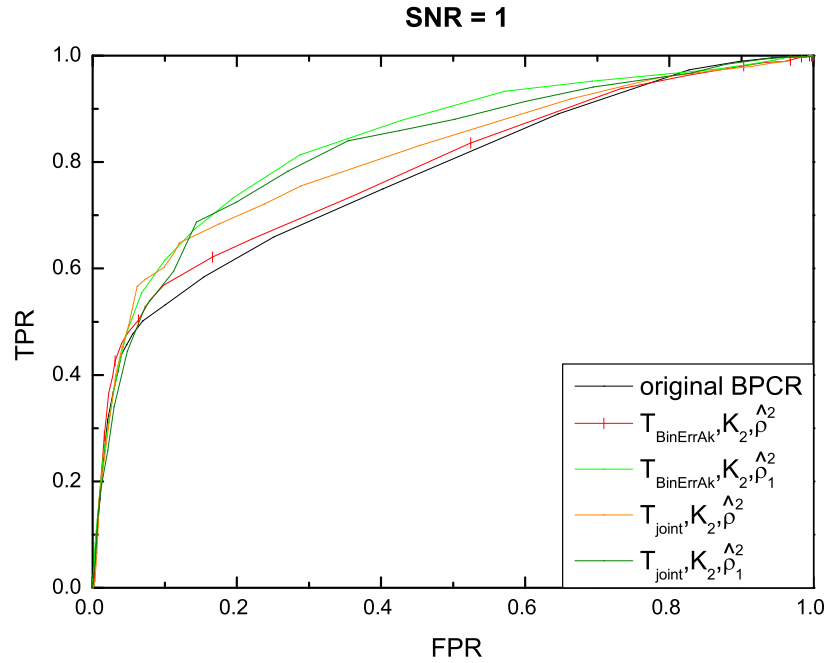


Figure S.12: ROC curves of several piecewise constant methods applied to dataset with SNR = 1. On this type of dataset with high noise, the modified BPCR with  $\hat{\rho}_1^2$ ,  $\hat{K}_2$  and  $\hat{T}_{\text{BinErrAk}}$  was the best performing method with respect to the ROC curve measure, because its curve was globally the highest one at the top left corner of the plot. GLAD, Rendersome and BioHMM were the worse methods in detecting the aberrations on this dataset.

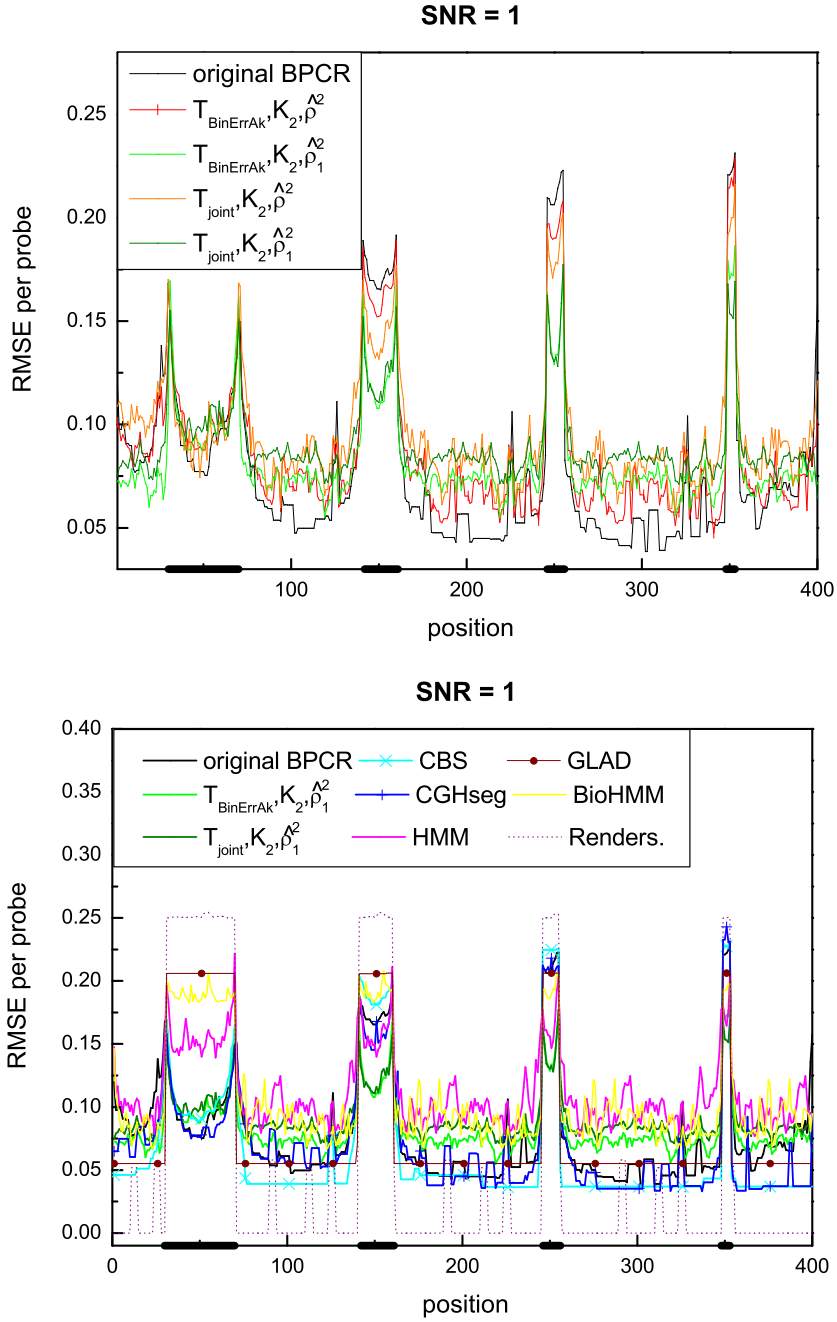


Figure S.13: RMSE of several piecewise constant methods applied to dataset with  $SNR = 1$ . The black segments on the horizontal axis correspond to the regions of aberration. On this very noisy dataset, the modified BPCR with  $\hat{\rho}_1^2$ ,  $\hat{K}_2$  and  $\hat{T}_{BinErrAk}$  always had a low RMSE per probe, even though its error was not the lowest one outside the aberrations and inside the first one. On the contrary, CBS and CGHseg had the lowest error in these regions, but the highest error inside the small aberrations. Hence, globally the modified BPCR with  $\hat{\rho}_1^2$ ,  $\hat{K}_2$  and  $\hat{T}_{BinErrAk}$  performed better than the other algorithms on this dataset, with respect to the RMSE measure.

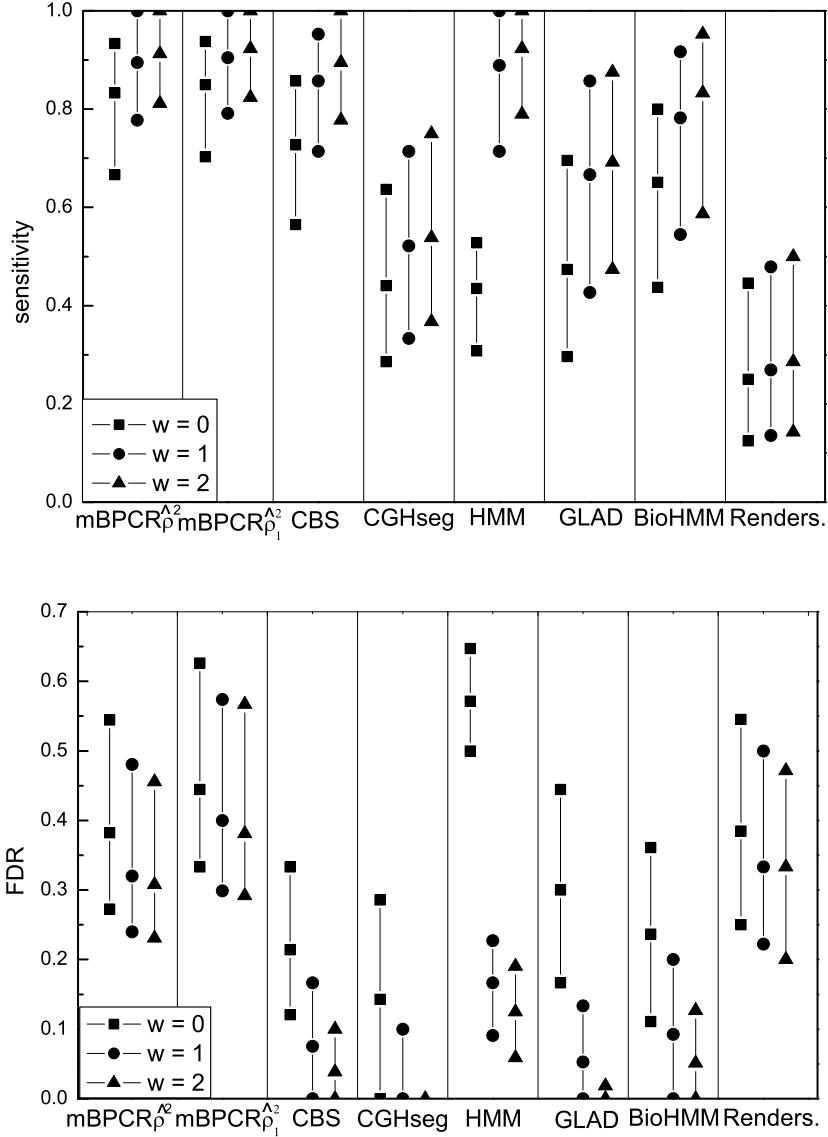


Figure S.14: Comparison of the sensitivity and FDR computed on the results obtained on dataset *Simulated Chromosomes*, using several piecewise constant methods. The  $mBPCR$  method (considering both  $\rho^2$  estimators) had the highest sensitivity, hence it was the method that determined the breakpoints location with highest precision. Nevertheless, it had a higher FDR than CBS.

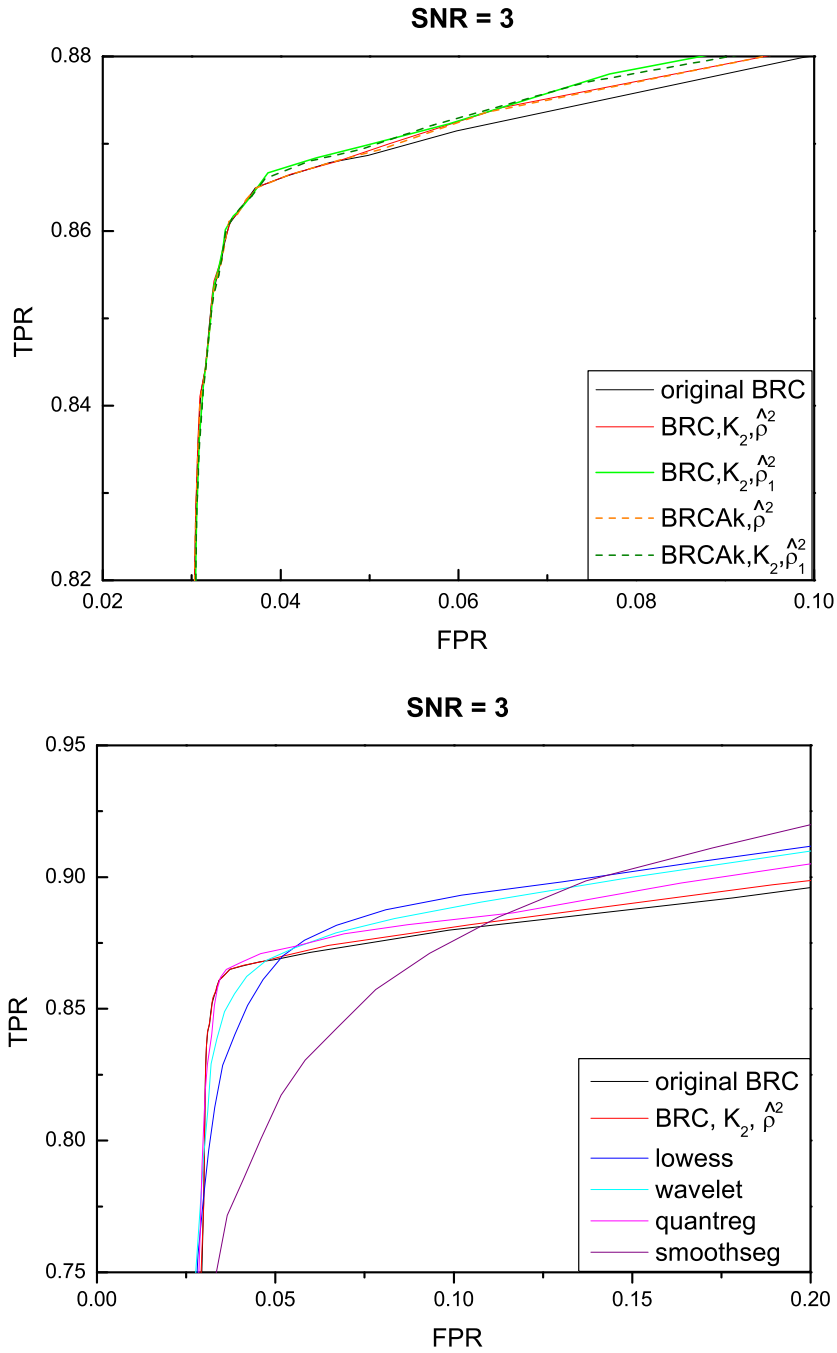


Figure S.15: Zoomed ROC curves of several smoothing methods applied to dataset with SNR = 3. The intersection among the ROC curves was due to the differences of the methods in the level estimation outside the aberrations. The more oscillating were the estimated curves in these regions, the closer were the corresponding ROC curves to the top side of the graph. In our case, an oscillating estimated profile is very different from the true one.

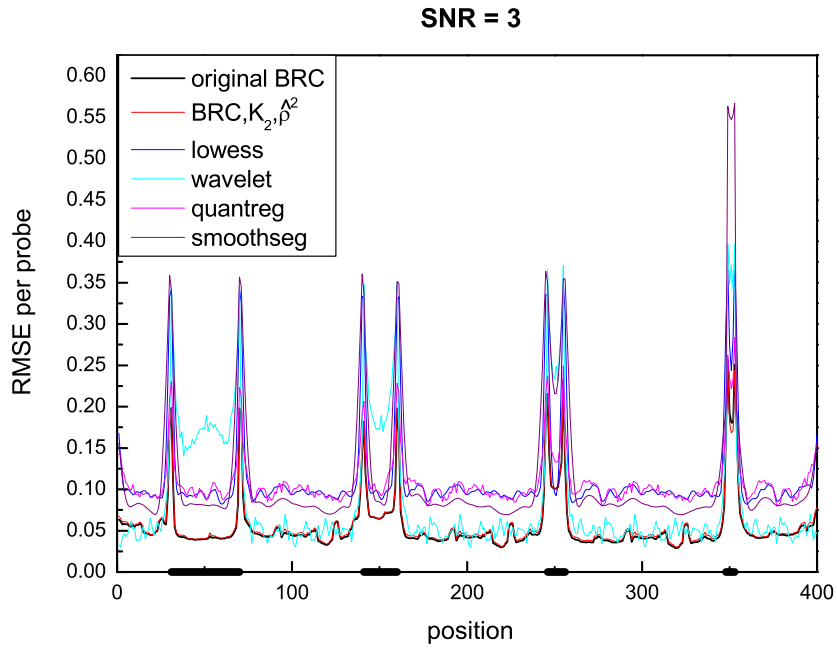
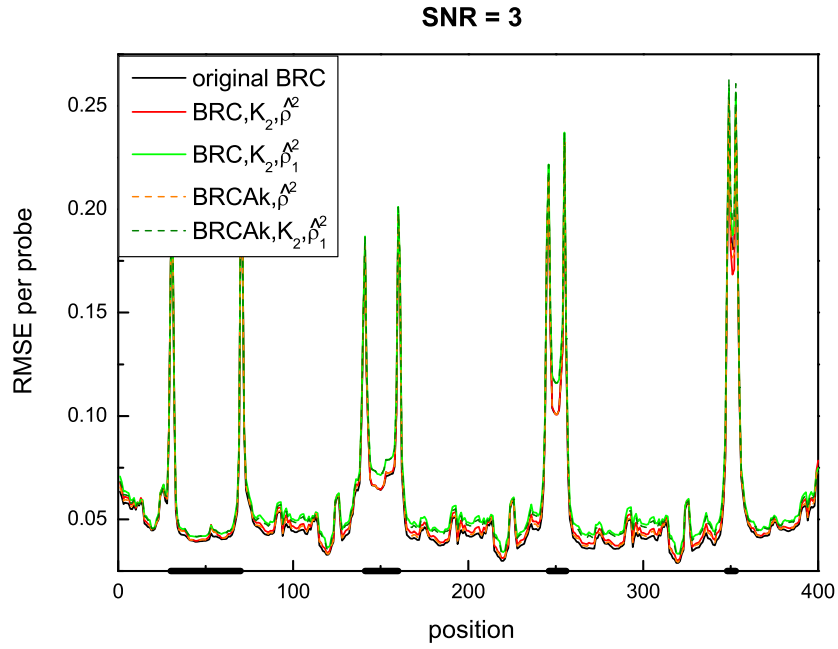


Figure S.16: RMSE of several smoothing methods applied to dataset with SNR = 3. The black segments on the horizontal axis correspond to the regions of aberration. On this dataset, both the original BRC and the version of BRC with  $\hat{K}_2$  and  $\hat{\rho}^2$  had everywhere the lowest error.

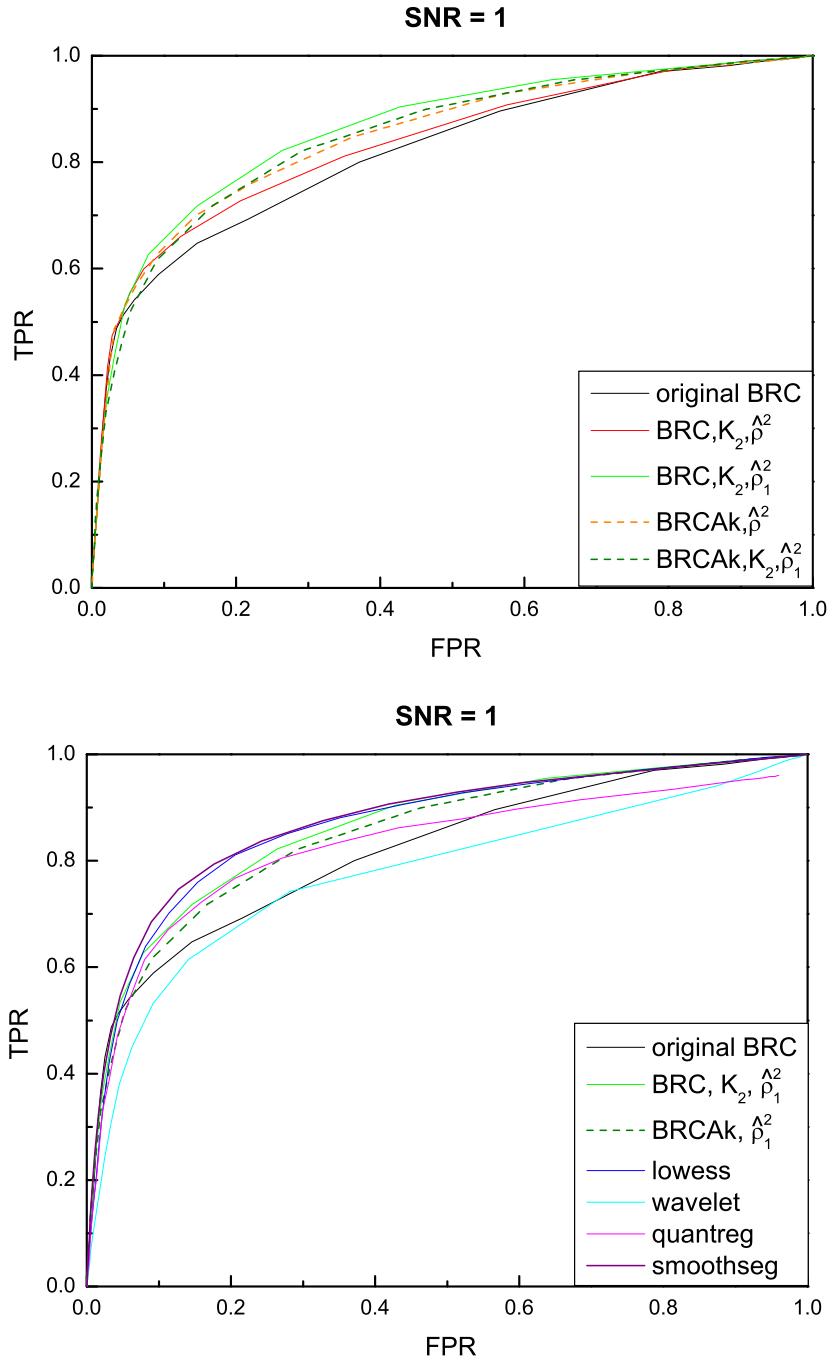


Figure S.17: ROC curves of several smoothing methods applied to dataset with  $\text{SNR} = 1$ . On this very noisy data, the methods smoothseg and lowess seemed to be the best ones, since their ROC curves were the highest at the top left corner of the plot. The third best method was BRC with  $\hat{K}_2$  and  $\hat{\rho}_1^2$ .

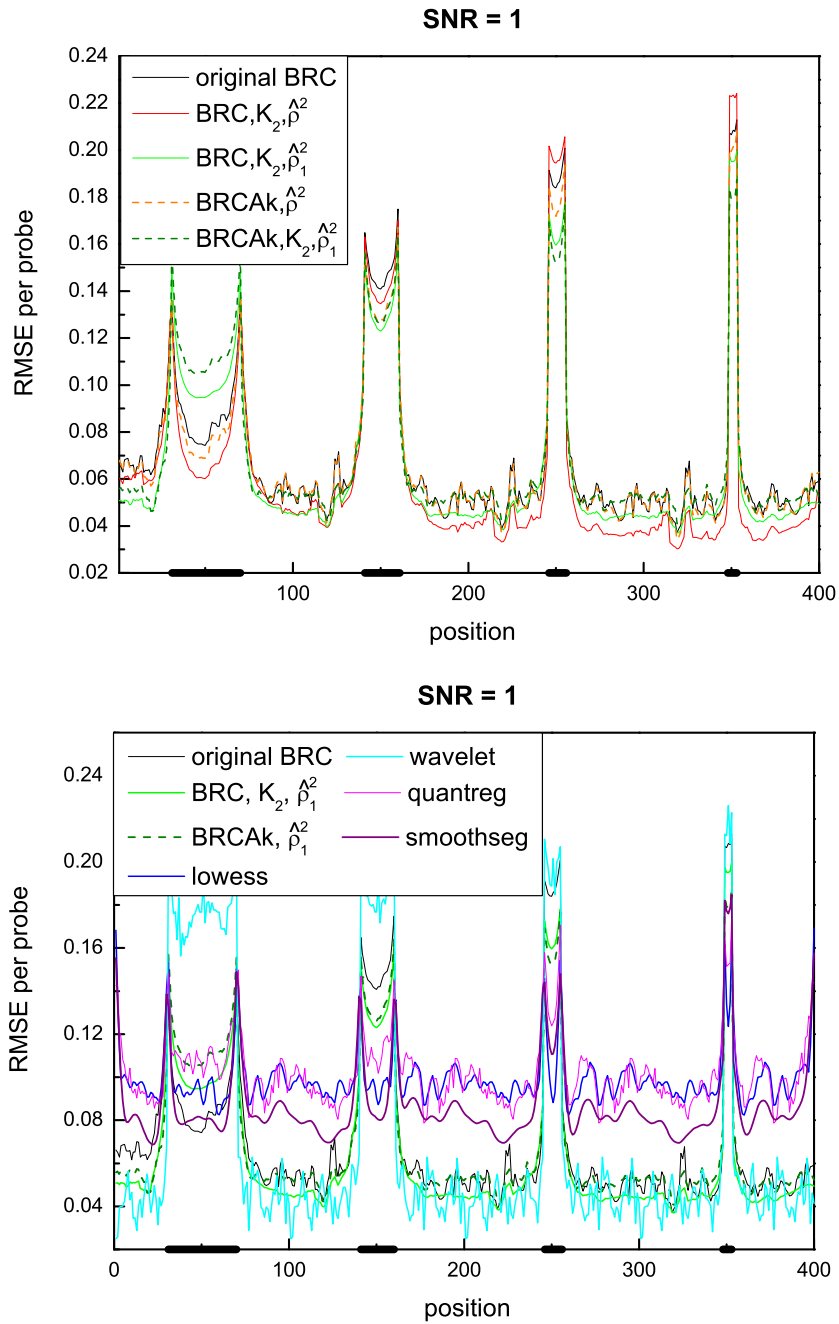


Figure S.18: RMSE of several smoothing methods applied to dataset with  $\text{SNR} = 1$ . The black segments on the horizontal axis correspond to the regions of aberration. The graphs show that the method lowess, quantreg and smoothseg had more or less the same error inside and outside the aberrations. Instead, the BRC version with  $\hat{K}_2$  and  $\hat{\rho}_1^2$  and BRCAk with  $\hat{\rho}_1^2$  had a very low error outside the aberrations and not the highest error inside them, thus globally they performed better than the other algorithms with respect to the RMSE measure.