附录 1：贝叶斯在线变点检测算法流程

算法 A.1. 贝叶斯在线变点检测算法

输入：时间序列 $x_1, x_2, \cdots, x_T$，风险函数 $\mathrm{H}(\tau) = 1/\lambda$，超参数先验 v_{prior} 和 χ_{prior}。

输出：对 $t = 2, \cdots, T$，返回 r_t 的后验概率 $\mathrm{P}(r_t \mid \boldsymbol{x}_{1:t})$。

初始化 $\mathrm{P}(r_0 = 1) = 1, \mathrm{P}(r_0 \neq 1) = 0, v_1^{(0)} = v_{\text{prior}}, \chi_1^{(0)} = \chi_{\text{prior}}$；

对 $t = 2, \cdots, T$ 执行：

计算 UPM：$\pi_t^{(r)} = \mathrm{P}(x_t \mid v_t^{(r)}, \chi_t^{(r)})$；

基于 CP 计算增长概率：$\mathrm{P}(r_t = r_{t-1} + 1, \boldsymbol{x}_{1:t}) = \mathrm{P}(r_{t-1}, \boldsymbol{x}_{1:t-1}) \pi_t^{(r)} \left[1 - \mathrm{H}(r_{t-1})\right]$；

基于 CP 计算变点概率：$\mathrm{P}(r_t = 0, \boldsymbol{x}_{1:t}) = \sum_{r_{t-1}} \mathrm{P}(r_{t-1}, \boldsymbol{x}_{1:t-1}) \pi_t^{(r)} \mathrm{H}(r_{t-1})$；

计算归一化概率：$\mathrm{P}(\boldsymbol{x}_{1:t}) = \sum_{r_t} \mathrm{P}(r_t, \boldsymbol{x}_{1:t-1})$；

计算 r_t 条件概率分布：$\mathrm{P}(r_t \mid \boldsymbol{x}_{1:t}) = \mathrm{P}(r_t, \boldsymbol{x}_{1:t}) / \mathrm{P}(\boldsymbol{x}_{1:t})$，记录当前的概率值；

更新统计量：$v_{t+1}^{(0)} = v_{\text{prior}}, \chi_{t+1}^{(0)} = \chi_{\text{prior}}, v_{t+1}^{(r+1)} = v_t^{(r)} + 1, \chi_{t+1}^{(r+1)} = \chi_t^{(r)} + u(x_t)$；

附录 2：贝叶斯在线变点检测采用 Gamma 分布假设的示例

在基于贝叶斯在线变点检测框架的计算中，若假设观测数据产生于 Gamma 分布 $\mathrm{Gamma}(a, b)$，根据 Gamma 分布的指数分布形式，其密度函数为

$$\mathrm{f}(x \mid a, b) = \frac{b^a}{\Gamma(a)} x^{a-1} \exp(-bx) = \frac{1}{\Gamma(a)} \exp\{\eta^{\mathrm{T}} u(x) - A(\eta)\}$$

其中参数 η 为 $(a-1, b)$，$A(\eta) = \ln \Gamma(a) - a \ln b$，对应的充分统计量为 $u(x) = (\ln x, -x)$。假定参数中 a 的先验分布为泊松分布 $\mathrm{Poisson}(\mu_0)$，b 的先验分布为 Gamma 分布 $\mathrm{Gamma}(\alpha_0, \beta_0)$，则可得关于 η 的超参数组合 χ 和 v 分别为 $(\ln \mu_0, -\beta_0)$ 和 $a^{-1}(\alpha_0 - 1)$。由逐点递推公式可得，在得到新观测值 x 后的更新公式为：

$$\mu_{t+1} = \mu_t x, \quad \beta_{t+1} = \beta_t + x, \quad \alpha_{t+1} = \alpha_t + a,$$

其中 α_t 的更新涉及到尺度参数 a 的取值，计算中可用 t 时刻 a 的后验估计替代。将依据更新计算得到的 $(\mu_t, \beta_t, \alpha_t)$ 作为 $v_t^{(r)}$、$\chi_t^{(r)}$ 带入贝叶斯在线变点检测算法可计算 UPM，实现变点检测。值得注意的是，Gamma 分布中尺度参数的选择将决定分布函数的形状，对模型效果具有较大影响，因此如何选择合适的先验超参数是实践中需要重点关注的课题。

附录 3：带有超参数更新的贝叶斯在线变点检测算法流程

算法 A.2. 带有超参数更新的贝叶斯在线变点检测算法

输入：长度为 T' 的观测序列 $x_1, x_2, \cdots, x_{T'}$，风险函数 $\mathrm{H}(\tau)=1/\lambda$，超参数先验 v_{prior} 和 χ_{prior}，

迭代次数 n，学习率 δ，$(R_0, \partial_h R_0, \partial_m R_0) \leftarrow (1,0,0)$。

输出：更新后的超参数 $\Theta=(\theta_h, \theta_m)$。

初始化 $\mathrm{P}(r_0=1)=1, \mathrm{P}(r_0 \neq 1)=0$，$v_1^{(0)}=v_{\text{prior}}$，$\chi_1^{(0)}=\chi_{\text{prior}}$；

对 $t=2,\cdots,T'$ 执行：

计算 UPM：$\pi_t^{(r)}=\mathrm{P}(x_t \mid v_t^{(r)}, \chi_t^{(r)})$；

计算增长概率：$\mathrm{P}(r_t=r_{t-1}+1, \boldsymbol{x}_{1:t})=\mathrm{P}(r_{t-1}, \boldsymbol{x}_{1:t-1})\pi_t^{(r)}\left[1-\mathrm{H}(r_{t-1})\right]$；

计算 θ_h 偏导数：$\partial_h R_t[2:t+1] \leftarrow \pi_t^{(r)} \partial_h R_{t-1}[1-\mathrm{H}(r_{t-1})]-R_{t-1}\partial_h \mathrm{H}(r_{t-1})$；

计算 θ_m 偏导数：$\partial_m R_t[2:t+1] \leftarrow [1-\mathrm{H}(r_{t-1})][\partial_m R_{t-1}\pi_t^{(r)}+R_{t-1}\partial_m \pi_t^{(r)}]$；

计算变点概率：$\mathrm{P}(r_t=0, \boldsymbol{x}_{1:t})=\sum_{r_{t-1}} \mathrm{P}(r_{t-1}, \boldsymbol{x}_{1:t-1})\pi_t^{(r)}\mathrm{H}(r_{t-1})$；

计算 θ_h 偏导数：$\partial_h R_t[1] \leftarrow \sum \pi_t^{(r)} \partial_h R_{t-1}\mathrm{H}(r_{t-1})+R_{t-1}\partial_h \mathrm{H}(r_{t-1})$；

计算 θ_m 偏导数：$\partial_m R_t[1] \leftarrow \sum \mathrm{H}(r_{t-1})[\partial_m R_{t-1}\pi_t^{(r)}+R_{t-1}\partial_m \pi_t^{(r)}]$；

计算归一化概率：$\mathrm{P}(\boldsymbol{x}_{1:t})=\sum_{r_t} \mathrm{P}(r_t, \boldsymbol{x}_{1:t-1})$；

计算 r_t 条件概率分布：$\mathrm{P}(r_t \mid \boldsymbol{x}_{1:t})=\mathrm{P}(r_t, \boldsymbol{x}_{1:t})/\mathrm{P}(\boldsymbol{x}_{1:t})$，令 $R_t=\mathrm{P}(r_t \mid \boldsymbol{x}_{1:t})$；

更新统计量：$v_{t+1}^{(0)}=v_{\text{prior}}, \chi_{t+1}^{(0)}=\chi_{\text{prior}}, v_{t+1}^{(r+1)}=v_t^{(r)}+1, \chi_{t+1}^{(r+1)}=\chi_t^{(r)}+u(x_t)$；

令 $\mathrm{P}(\boldsymbol{x}_{1:T'}) \leftarrow \sum R$，$\partial \mathrm{P}(\boldsymbol{x}_{1:T'}) \leftarrow \left(\sum \partial R_T, \sum \partial_m R\right)$；

利用 $\partial \mathrm{P}(\boldsymbol{x}_{1:T'})$ 对超参数进行梯度下降更新：$\hat{\Theta} \leftarrow \hat{\Theta}-\delta \cdot \partial \mathrm{P}(\boldsymbol{x}_{1:T'})$；

重复上述梯度计算、更新过程 n 轮次，返回 n 次迭代更新后超参数 Θ 的结果。

附录 4：PELT 算法过程

算法 A.3. PELT 算法

输入：时间序列 $x_1, x_2, \cdots, x_T$，损失函数 $\mathrm{C}(\cdot)$，惩罚项系数 β_p，常数 K_p。

输出：变点集合 CP。

初始化 $\mathrm{F}(0) = -\beta_p$，$CP = \varnothing$，$R_1 = \{0\}$；

对于 $t = 1, \cdots, T$：

计算 $\tau^1 = \arg\min_{\tau \in R_t} \left[\left\{ \mathrm{F}(\tau) + \mathrm{C}\left(\boldsymbol{x}_{\tau+1:t} \right) \right\} + \beta_p f_p(m) \right]$；

将 τ^1 加入 CP 中，即令 $CP = CP \cup \{\tau^1\}$；

令 $R_{t+1} = \{\tau \in R_t \cup \{t\} : \mathrm{F}(\tau) + \mathrm{C}\left(\boldsymbol{x}_{\tau+1:t} \right) + K_p \le \mathrm{F}(t)\}$；

返回变点集合 CP。

其中常数 K_p 根据惩罚项的设置确定，具体可见 Killck 等（2011）在 PELT 方法中的讨论。

附录 5：带有滑动窗口的贝叶斯在线变点检测方法

算法 A.4. 带有滑动窗口的贝叶斯在线变点检测方法

输入：时间序列 $x_1, x_2, \cdots, x_T$，W，D

输出：变点集合 CP。

1.初始化变点序列 $CP=\varnothing$， $t_{\text{start}}=0$；

2.当 $t_{\text{start}}<T$ 时，执行：

3. 当 $t_{\text{start}}+\alpha<T$ 时：

4. 对序列 $\{x_t : t=t_{\text{start}},\cdots,t_{\text{start}}+\alpha\}$，利用 PELT 算法进行变点检测；

5. 若发现变点 τ，则加入 CP，并令 $t_{\text{start}}=\tau$，返回步骤 2，若未发现变点，进入步骤 6；

6. 在序列 $\{x_t : t=t_{\text{start}},\cdots,t_{\text{start}}+\alpha\}$ 上运行带有超参数更新的贝叶斯在线变点检测算法，得到超参数取值 $\hat{\theta}$；

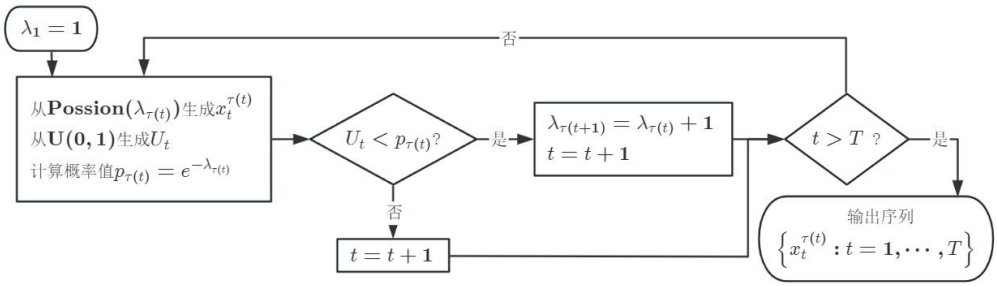
7. 令 $t_{\text{end}}=\min(t_{\text{start}}+W,T)$，在序列 $\{x_t : t=t_{\text{start}},\cdots,t_{\text{end}}\}$ 上运行贝叶斯在线变点检测算法（以 $\hat{\theta}$ 作为超参数的输入值）；

8. 若发现变点 τ，则加入 CP，并令 $t_{\text{start}}=\tau$，否则令 $t_{\text{start}}=t_{\text{start}}+D$，返回步骤 2；

9. 否则：

10. 在序列 $\{x_t : t=t_{\text{start}},\cdots,T\}$ 上运行 PELT 算法，将发现的变点加入 CP；

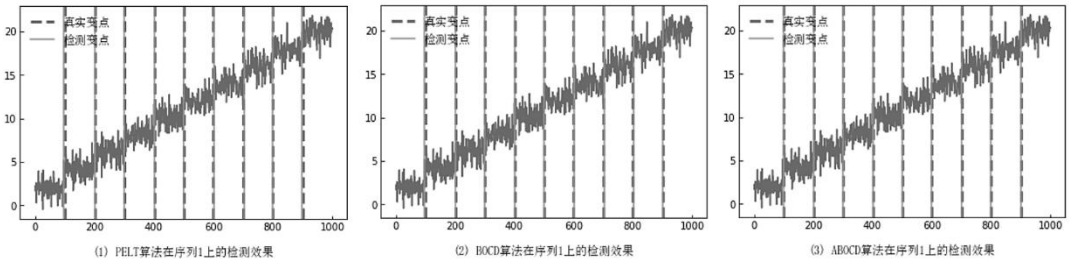
算法结束，返回变点集合 CP。



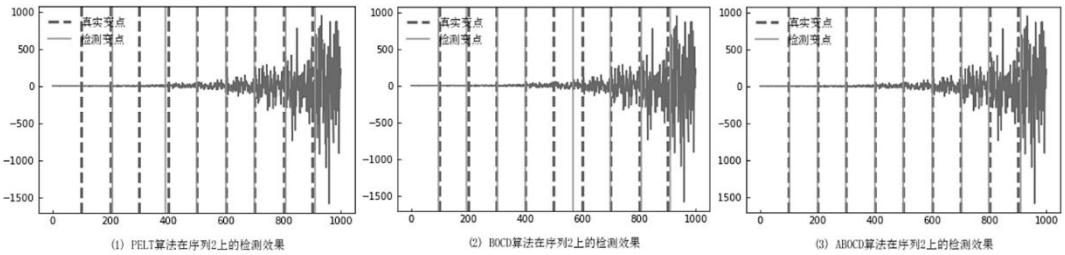
附图 1　模拟数据生成机制

附表 1　　不同惩罚项设置下变点检测效果

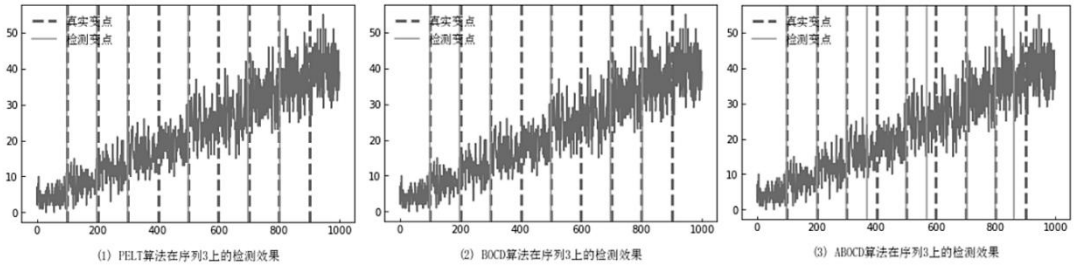
损失函数：L_1	覆盖度				改进的 F_1				
惩罚系数	序列 1	序列 2	序列 3	序列 4	惩罚系数	序列 1	序列 2	序列 3	序列 4
0.1	0.973	0.922	0.868	0.768	0.1	0.903	0.894	0.890	0.793
0.5	0.942	0.904	0.850	0.768	0.5	0.864	0.861	0.880	0.793
1	0.924	0.898	0.850	0.667	1	0.828	0.861	0.835	0.746
5	0.989	0.922	0.750	0.664	5	0.947	0.930	0.860	0.774
10	0.989	0.928	0.857	0.796	10	0.947	0.966	0.893	0.812
20	0.989	0.928	0.857	0.796	20	0.947	0.966	0.893	0.812
损失函数：L_2	覆盖度				改进的 F_1				
惩罚系数	序列 1	序列 2	序列 3	序列 4	惩罚系数	序列 1	序列 2	序列 3	序列 4
0.1	0.956	0.894	0.864	0.705	0.1	0.936	0.871	0.860	0.764
0.5	0.939	0.896	0.842	0.506	0.5	0.895	0.825	0.787	0.764
1	0.880	0.875	0.723	0.642	1	0.910	0.821	0.789	0.739
5	0.933	0.876	0.785	0.756	5	0.926	0.839	0.811	0.739
10	0.979	0.876	0.830	0.768	10	0.940	0.933	0.833	0.788
20	0.979	0.892	0.830	0.751	20	0.937	0.933	0.860	0.788



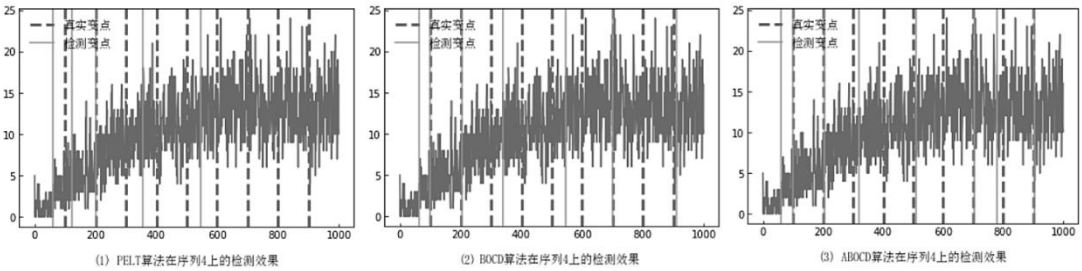
附图 2　模拟实验中三种算法在序列 1 上检测效果



附图 3 模拟实验中三种算法在序列 2 上检测效果



附图 4 模拟实验中三种算法在序列 3 上检测效果



附图 5 模拟实验中三种算法在序列 4 上检测效果

附录 6：模拟实验中序列检测过程

图 A.1、图 A.2、图 A.3 针对正文模拟实验中序列 2、序列 3、序列 4 的模拟数据分别展示了变点检测的具体过程。对每个模拟设置，子图(a)展示了模拟产生的时间序列，子图(b)为每个时间点 t 上 r_t 取值的概率分布，颜色越深表明对应的 r_t 值概率密度越大。子图(c)和(d)分别为超参数中心 μ_0 和精度 Λ_0 随着滑动窗口推进的更新情况。与正文中序列 1 上的结论类似，在各组模拟数据上，本文提出的 ABOCD 方法中超参数的更新与序列的演化设置较为一致，计算得到的 r_t 后验分布与原始序列的变点信息相吻合，从而及时识别变点。

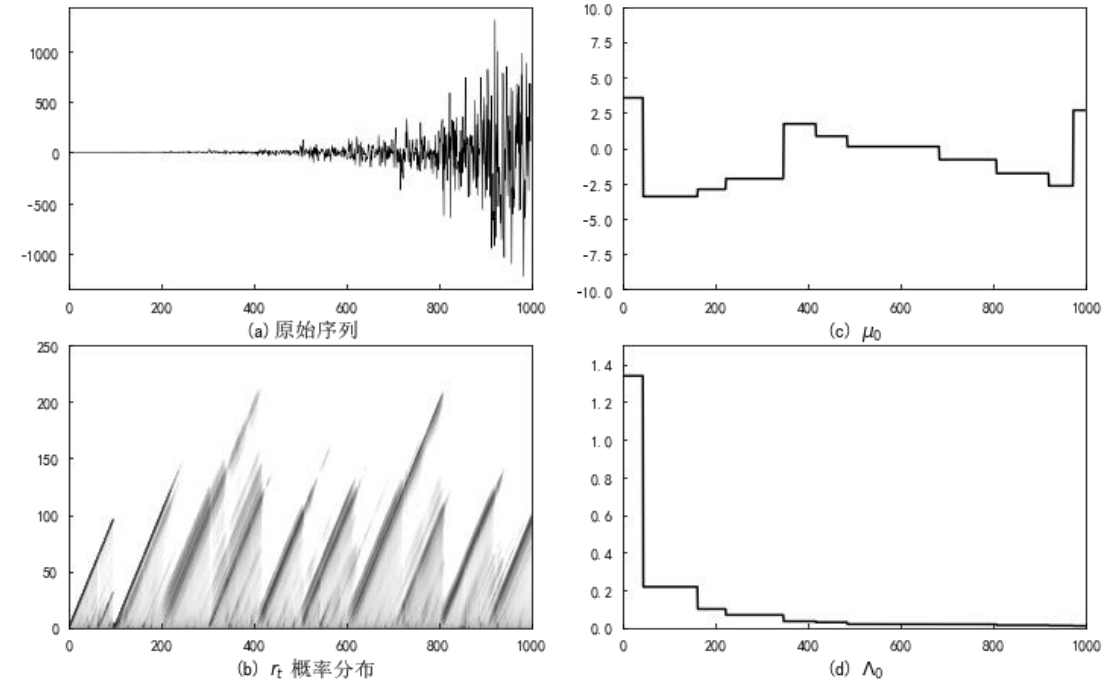


图 A.1 序列 2 检测过程：原始序列、r_t 概率分布，及超参数更新情况

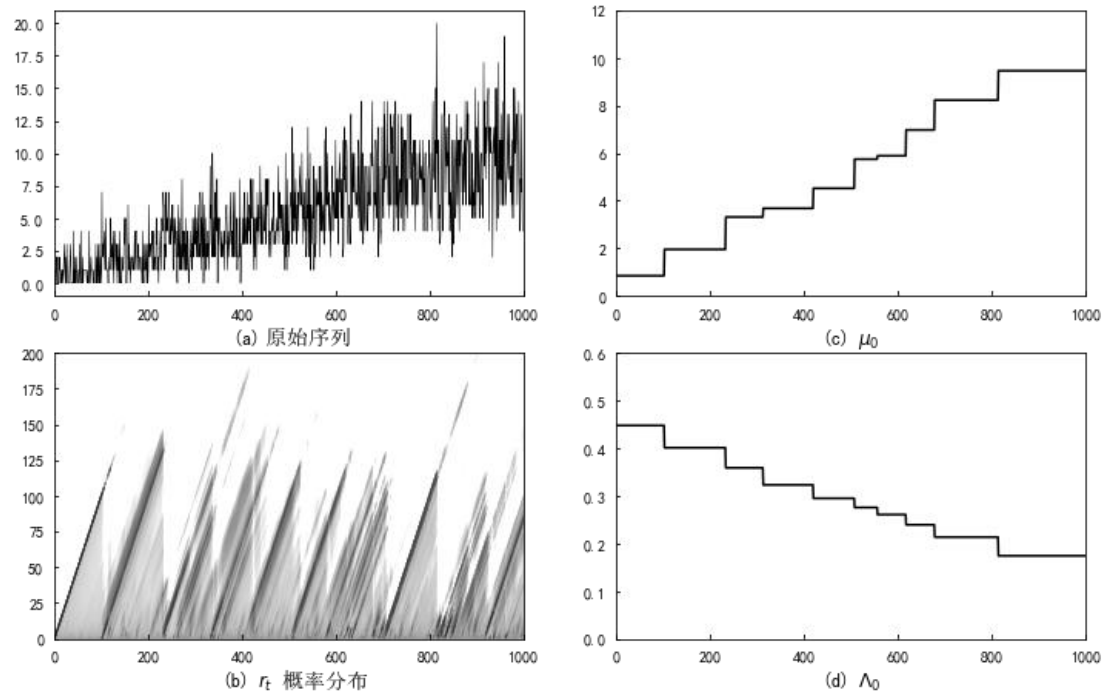


图 A.2 序列 3 检测过程：原始序列、r_t 概率分布，及超参数更新情况

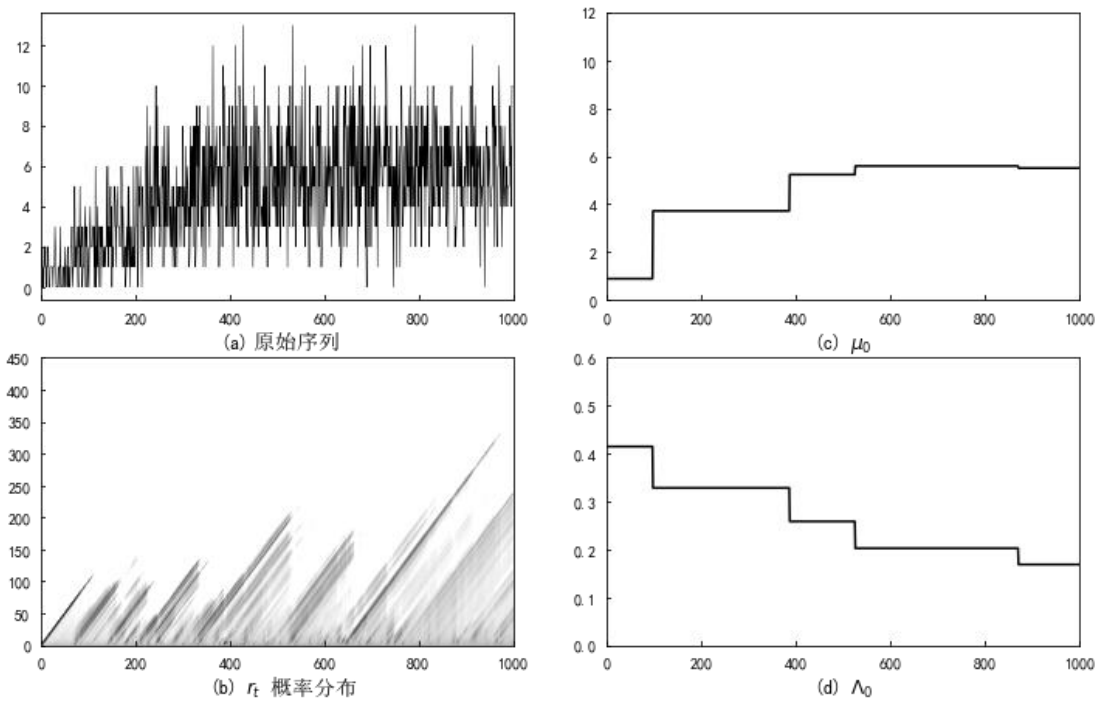
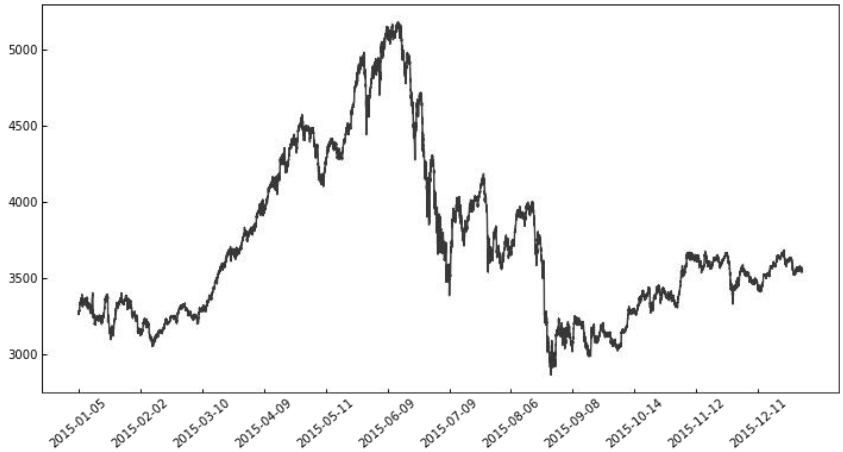
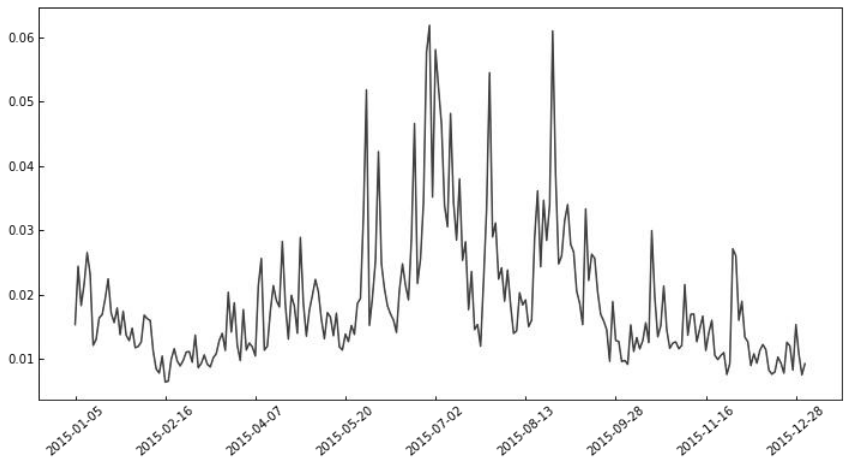


图 A.3 序列 4 检测过程：原始序列、r_t 概率分布，及超参数更新情况



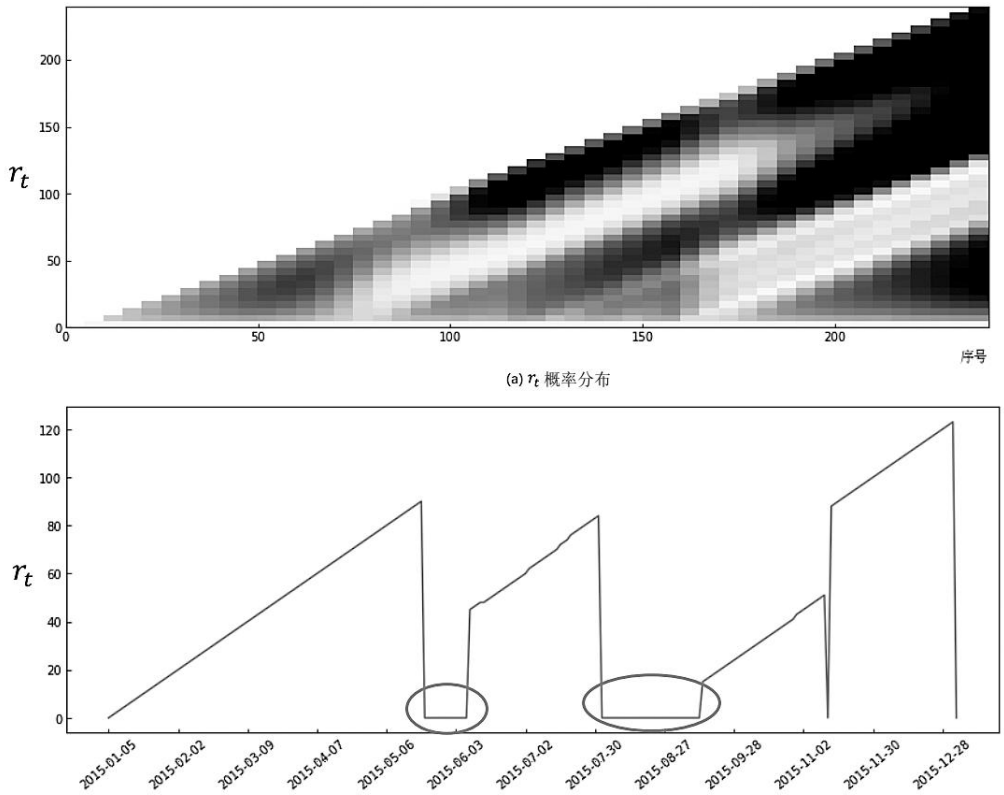
附图 6　2015 年上证综指价格指数序列



附图 7　2015 年上证综指日内 RV 序列

附录 7：BOCD 算法在上证综指序列上的检测效果

首先在每日收盘价格序列上直接运行 BOCD 算法作为基准，检测效果如图 A.4 所示，其中 BOCD 算法的风险函数中常数 $\lambda = 1/100$。图 A.4 中(a)是 BOCD 算法检测过程中每一个时间点上 r_t 取值的概率分布，颜色越浅表示概率越大，图 A.4 中(b)是根据(a)返回的每一时间节点取最大概率值对应的 r_t 值，当 r_t 值为 0 时，即认为当前时间点为变点。从图中可以看出，当时间序列波动较为剧烈时，BOCD 算法无法有效发现变点，例如在两圆圈标记处，表明这一段序列上处处为变点，而结合附图 6 和附图 7 的分析可知，这两段序列正好对应波动率较大、序列的剧烈上升或下降时期。这样的检测结果无法指导实际应用，因为当每一天都被标记为变点时，也就失去了变点用于划分不同阶段的意义。



(a) r_t 概率分布

(b) 每一时刻最大概率r_t的取值

图 A.4　BOCD 算法在收盘价格序列上的检测效果（$\lambda = 1/100$）

接下来考虑调整风险函数，令 $\lambda = 1/500$，观察结果如图 A.5 所示。由当风险函数取值设置较小时，能检测出的变点数量锐减，在整个序列上只能检测出 3 个变点，分别对应 5 月中下旬股市大幅波动即将开始之际、9 月中下旬股市大幅波动大体结束进入震荡期以及 11 月初股市恢复自我调整功能进入稳定期三个阶段，虽然能检测出这些关键节点，但与直观分析的结果相去甚远。在指导实际金融市场活动时，需要更为细致的结果。

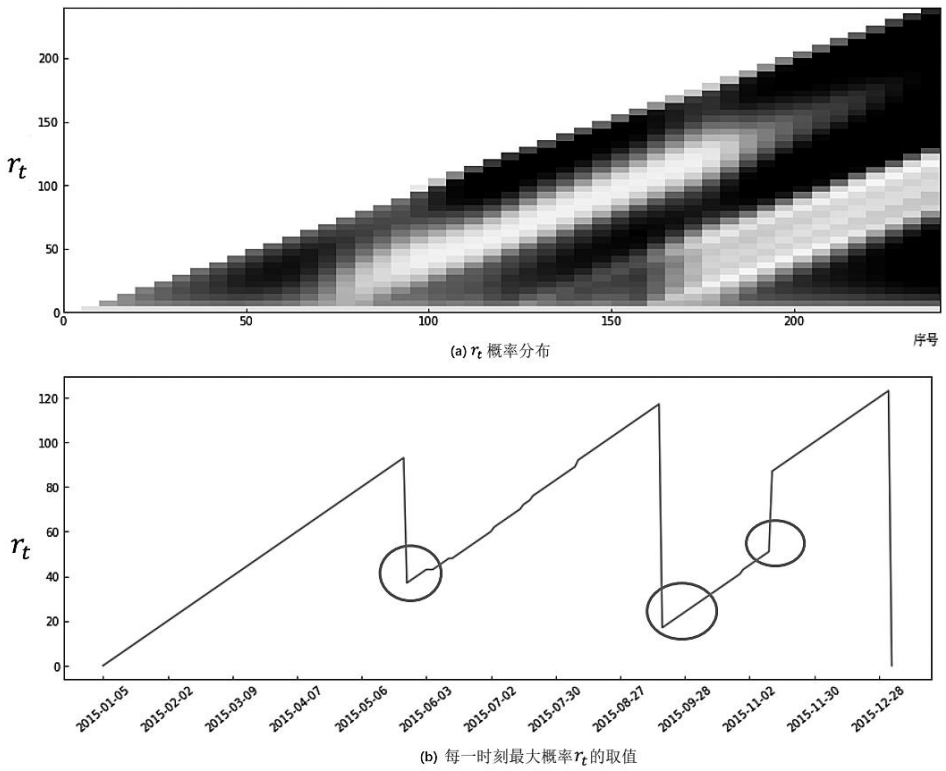


图 A.5　BOCD 算法在收盘价格序列上的检测效果（$\lambda = 1/500$）

附表 2　　共同识别的变点及收盘价序列检测的补充变点

收益率序列检测变点	收盘价序列检测变点	说明
	3 月 9 日	进入一轮持续上涨
	4 月 23 日	由调整转入上涨
5 月 20 日	5 月 26 日	收益率序列检测提前 6 个交易日
	6 月 30 日	出现止跌信号
8 月 11 日	8 月 20 日	收益率序列检测提前 9 个交易日
9 月 25 日	10 月 9 日	收益率序列检测提前 7 个交易日
	11 月 5 日	从上涨进入调整修复